

ORIC OWNER

ISSUE

DEC/JAN '84



INSIDE THIS ISSUE

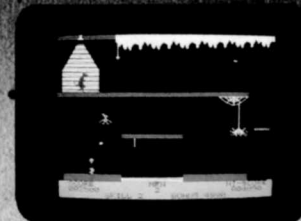
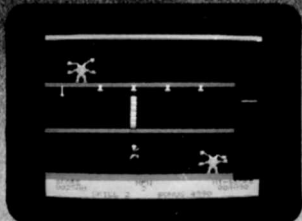
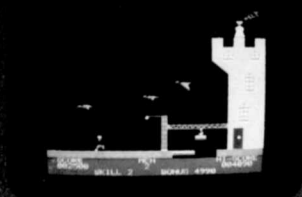
Win a printer - superb competition

Pull out ORIC circuit diagram

Forth: Money Manager

Biggest Issue Ever! 88 pages!!!

IJK Software and the ORIC-1 Micro bring you more exciting arcade action from the Xenon series...



From the author of the highly acclaimed Xenon-1 comes the second in the Xenon series. This game continues the high standard of excellence set by its predecessor. The amazing super high resolution graphics make this 100% machine code arcade game a must for your collection.

Following their defeat at the hands of the Xenon fleet the Zorgons have captured the Xenon princess Roz, and have imprisoned her in their castle. You are commissioned to rescue her by scouring the four corners of the Zorgon Empire to capture the magic stones. These stones, guarded by the Quadnogs, Terrapods and many other strange beasts, are needed to bridge the bottomless chasm surrounding the castle, enabling you to achieve your goal.

Each one of the many varied stages in this scintillating mission will test your arcade ability as never before.

FEATURES INCLUDE:-

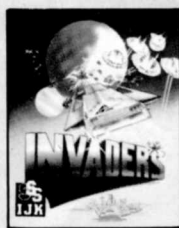
- 100% MACHINE CODE
- SUPERB SOUND EFFECTS
- SUPER-SMOOTH HI-RES GRAPHICS
- MULTI-STAGE ARCADE ACTION
- SKILL LEVELS
- HALL OF FAME

For the 48K ORIC-1 £8.50 inc.

OTHER TITLES IN THE IJK SOFTWARE ORIC-1 RANGE...



XENON-1
100% M/C arcade
game for the
48K ORIC-1
£8.50 inc.



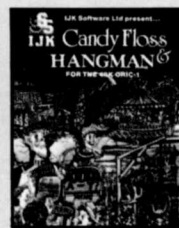
INVADERS
Machine code
arcade game for
16K & 48K ORIC-1
£7.50 inc.



FANTASY QUEST
Intriguing
adventure for
48K ORIC-1
£6.50 inc.



REVERSE
Superb board
game for
48K ORIC-1
£6.50 inc.



**CANDYFLOSS &
HANGMAN**
Two top
educational
programs for
48K ORIC-1



**3D MAZE &
BREAKOUT**
Arcade action for
48K ORIC-1
£7.50 inc.

ALL PRICES FULLY INCLUSIVE OF VAT and P&P - NO MORE TO PAY

Our software has been officially approved by Oric Products International Ltd., and is available from branches of W.H. Smiths, Laskys, Computers for all, Zappo, Boots, John Menzies, The Spectrum Chain, many other leading department stores, hundreds of independent dealers nationwide, and in 23 countries across the world. You can also order direct from us - all advertised software is in stock now and will be despatched within 48 hours of receipt of order.



**IJK
Software
Limited**



24 HOUR ANSAFONE

9 King Street, Blackpool, Lancs.
Telephone (0253) 21555

WARNING: All software sold subject to IJK Software's standard conditions of sale and terms of trade, copies available on request.



ORIC OWNER

Issue 5 Dec/Jan

Features

Oric-Base revealed	page 16-17
Hints and Tips	18-21
Extracts from the Oric Handbook	23-30
Oric-Copy	72-73
Circuit Diagram	
Pull-Out	44-45

Regulars

Editor's Comment	5
Captain Tanex	13
I/O Page	78
Subscription Form	83

News

News Brief	6-7
Software Scan	8

Software

Simpdis	10-12
Connect-4	14-15
Take Home Pay	
Calculation	31-32
And Forthly . . .	33-37
Rubik Cube	38-39
Missionaries and Cannibals	49
Cinderella	50-55
Scott Joplin's 'Entertainer' Rag	56-57
Grand Prix	58
Counting Shapes	59-60
A Question of Death	61-64
MC Load	65
Code Master Deluxe	68-71
Simple Simon	74-76
Maths	77
Quickies	40 + 79-82
Cassette File Manager	84-85



Oric-Base revealed

The author of Oric-Base, Geoff Phillips, reveals more to us about the program



Captain Tanex

How much longer can our intrepid hero escape death? Will he complete his mission in time?



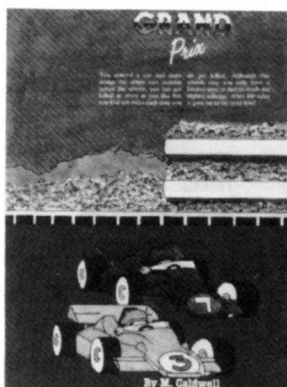
Take Home Pay Calculation

Is it really that little?! Try the program and find out



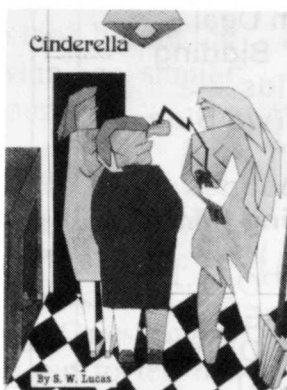
Hints & Tips

More recipes for you to cook up on your Oric!



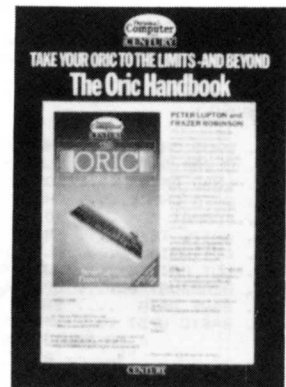
Grand-Prix

Become a formula 1 Grand Prix racing driver!



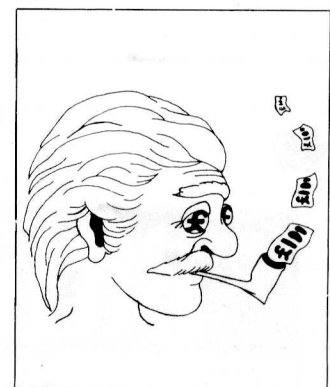
Cinderella

Will she find Prince Charming and marry him or will one of her ugly sisters get there first?



Extracts from the Oric Handbook

In this issue we feature the chapters on Communication and Programming



And Forthly . . . Money Manager

Another program for all you Forth enthusiasts



Competition!

Complete the crossword and caption and you could win yourself an Oric printer

Competition

Win an Oric Printer 66-67

Oric Owner Magazine

Editor: Paul B. Kaufman

Graphics & Feature Editor:

Carolyn Groeneveld

Administration: Carolyn

Groeneveld

Technical Consultant: Dr Paul Johnson

Printers: Heffers Printers Ltd, King's Hedges Road, Cambridge.

Photographic Equipment: Minolta

Oric Owner is published at bi-monthly intervals by Tansoft Ltd, Reg. No. 1632070

Units 1 & 2, Cambridge Techno Park, Newmarket Road, Cambridge. Advertising Rates are available on request.

© Tansoft Ltd 1984.

PEACH COMPUTERS LTD

192 GREENOCK RD., LARGS, Ayrshire, SCOTLAND. TEL: 0475 673766

FEACHBYTES

ON SALE WORLDWIDE SEE YOUR LOCAL AGENT OR DIRECT FROM US.
WRITE FOR FULL DETAILS OF OUR EXCITING RANGE OF SOFTWARE.
FOR EXAMPLE - -

RECORDS PB3 £10

Fast powerful low cost system for storing and easy access to vast quantities of data. Suitable for a wide variety of business educational home and scientific applications. Exploits the full editing facilities of the ORIC-1.

MENU

FUNCTION

P ... PROG CHECK CHECKS THE PROGRAM FOR ERRORS.
C ... CLEAR DATA CLEARS THE DATA STORE (30K BYTES).
L ... LOAD DATA LOADS IN DATA FILES FROM CASSETTE.
D ... DATA CHECK CHECKS THE DATA STORE FOR ERRORS.
F ... FORMAT 40/80 CHARS PER LINE ON PRINTER.
W ... WRITE TEXT WRITE AND EDIT TEXT TO VIDEO SCREEN.
E ... EDITORIAL FLIP THROUGH 30 SCREEN PAGES (30K).
O ... OUT PRINTER.... OUTPUTS SELECTED PAGE(S) TO PRINTER.
R ... READJUST DC.... UPDATES DATA CHECK AFTER ADDITIONS.
S ... SAVE DATA SAVES DATA FILES ON CASSETTES.
B ... BASIC UPDT UPDATES PROG CHECK AFTER CHANGES.
(NO SQUIGGLES OR MISPRINTS)

We specialise in fast efficient software for the ORIC-1
Full back up and support provided to all customers.
Our suite of software is steadily increasing with new and dramatic packages being developed. Keep in touch for news

(ALL PRICES FULLY INCLUSIVE)

ATTRACTIVE DEALER TERMS

ORIC Software

We have 68 programs available especially for the ORIC.

Airline £7.95 The Hobbit £14.95 Chess £9.95
Othello £4.95 Flight £7.95 Zodiac £6.90
PASE Joystick Interface £14.95

Books:

The Oric and How to get the most of it £5.95
The Oric-1 Book of Games £5.95
The Oric-1 Programmer £5.95
Oric Graphics and Sound £6.95
Oric Machine Code £6.95

Add 45p P&P to Books

P&P free on software. All prices inc. VAT.
Cheques/P.O. to Micro World.

For free descriptive software catalogue write to

Micro World
1 Lane Close,
London NW2 6QZ

BRIDGE

ORIC 48

ICB Bridge is a sophisticated bridge game for a single player, the other three hands are bid and played by the computer.

Features: Random Deal
Natural "Acol-type" Bidding

With: Jump Bids

Strong Two Bids
Stayman

Advanced Play routines
Only £9.85 including P&P

Send Cheque or PO to

ICB ENTERPRISES
33 BURNT STONES DRIVE
SHEFFIELD S10 5TT

Tel: (0742) 302050

DEALER ENQUIRIES WELCOME

THE ORIC-1 COMPANION

by Bob Maunder

ISBN 0 907211 03 8.

173 pages.

Price £6.95



A thorough reference guide for those Oric-1 owners who want to get to know and use the machine in depth.

Section 1: BASIC Summary
Section 2: Keywords Guide
Section 3: Screen Display
Section 4: Program Organisation
Section 5: The ROM
Appendices including The MCP-40 Printer

This is the latest in the *Companion* series, acclaimed by *Your Computer* as 'far and away the best for serious users'.

AVAILABLE FROM GOOD BOOKSHOPS OR

Send your cheque for £6.95
(UK p&p free) to

LINSAC

68 Barker Road,
Middlesbrough TS5 5ES

INCLUDES FULL
ROM DIS-
ASSEMBLY

Editor's Comment

By Paul B. Kaufman



Congratulations are the order of the day to Oric for winning the Best Micro of the Year award in France. The micro is relatively new to the French public and so this award takes on a particular significance.

Unlike the British public the French have no prejudices for or against any particular manufacturers or existing user base. Thus comparisons were strictly done on a specification and price against price basis. It is therefore rather gratifying to see that when the Oric is compared with other micros, such as the ZX-Spectrum, an unbiased public makes the Oric its first choice.

One interesting reason that the Oric has beaten the Spectrum in France is due to the Spectrums

lack of an RGB output. At first this may not seem particularly significant but when you realise that the French TV system (SECAM) will not work with computers using the UKs (PAL) system and that by coincidence most French TVs are equipped with RGB sockets, then you can begin to realise one of the Oric's big advantages in France. Let's hope its not too long before the Oric starts winning similar awards in this country.

On Friday the 14th of October, about 8 o'clock in the morning just as I was trying to stop the cat finishing off my breakfast, Chris Shaw, one of the Orics engineers, phoned me up to tell me that Kenure Plastics, Orics main factory and assembly plant, had burnt down the previous night and could I bring my cassette

recorder in. 'Oh no', thought I, 'they don't want me to duplicate 20,000 Welcome-to-Oric tapes by hand!' When I reached the office, rather late as I got lost on the way to work, Chris informed me that within about four hours of the factory burning down they had managed to set up a new plant and could they have some more copies of the special test tapes that are used to align the Orics. This particularly impressed me as considering the scale of the disaster I expected them to be out of production for several weeks – Well done Oric.

Incidentally the reason that I got lost going to work is that the previous day we moved offices. Our new address is Units 1 and 2, Techno Park, Newmarket Road, Cambridge. Tel: Teversham 2261-2-3-4.

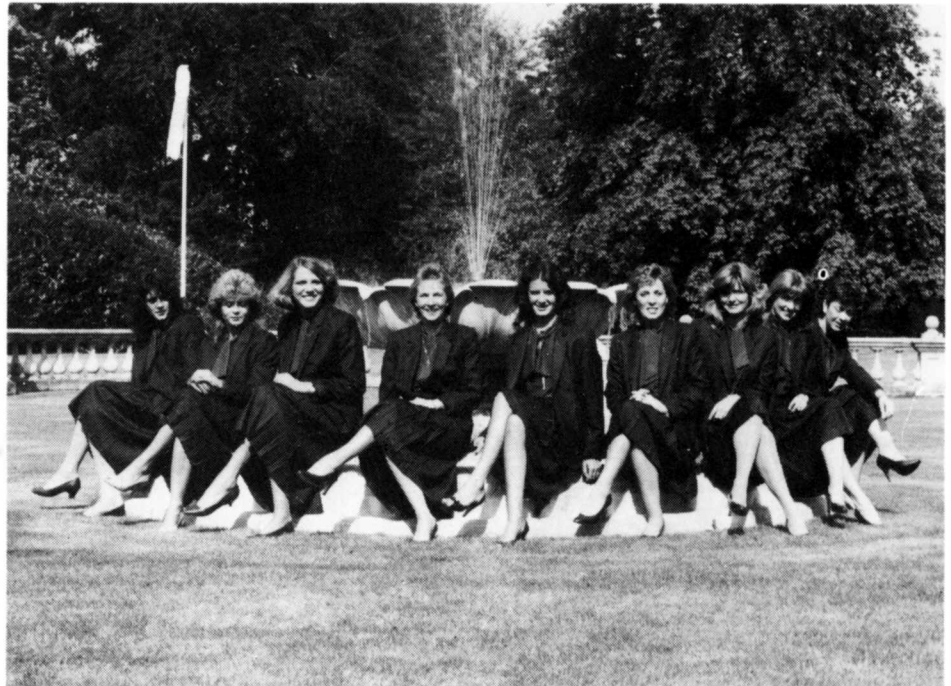


News Brief

Oric Angels to aid dealers

Oric Products International are putting a team of 'Angels' on the road as part of their campaign to give after sales service and support to their 1,000 dealer outlets.

The girls are intended to be merchandisers and not sales representatives. They will be there to help dealers and not to make sales.



The 14 attractive Angels – dressed in navy and burgundy uniform – will be based around all the major towns in the UK. Their call cycles will be carefully organised so that they will see every dealer at least once a fortnight.

Nine of the team are already on the road and the rest are receiving training at the company's headquarters in Ascot. They were carefully chosen from among 400 applicants for their enthusiasm and interest in computers.

Detailed call sheet information will provide useful feedback to Oric as to how they can best support dealers in the future.

Techno Park Move

As you may or may not know, we have now moved to a new block of offices, which we will be sharing with Oric Research.

All orders, technical enquiries and samples should in future be sent to:

Units 1 and 2,
Techno Park,
Newmarket Road,
Cambridge.

Tel: (02205) 2261-2-3-4

We also have a new member of staff: Debbie, our receptionist, will be here to help with order queries and help with the Sales Department.

*A Merry Christmas
to all our readers!
From all the
staff of Tansoft
and Oric!*

Exhibitions



P.C.W. Show

Tansoft made its first appearance at the PCW show at the Barbicon centre in October.

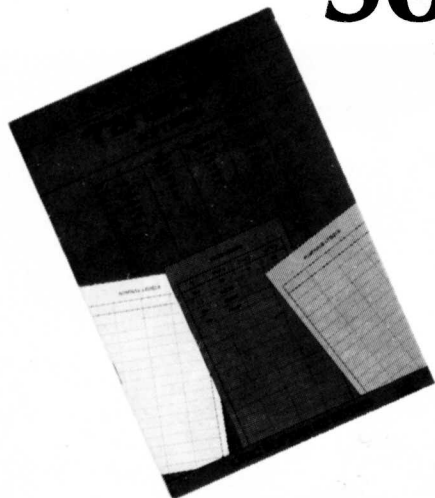
After a small argument with a neighbouring stand, the rest of the P.C.W. Show was very successful. A lot of interest was shown in our forthcoming products: Defence Force, Rat-Splat, the Language Masterclass Series and of course the Hobbit. We will aim to attend again next year and we hope to see as much interest then.

Your Computer Christmas Fair (at Wembley)

Tansoft will be represented at the Christmas Fair (15-18th Dec). If you have any queries or problems or just want to try a game of Defence Force, pop along and see us there!

Software Scan

By P. B. Kaufman



For business users who have become fans of Oric-Base comes a new offering from Tansoft – Oric-Calc. This has been written by the author of Oric-Base, Geoff Phillips, and is well up to his usual standard. Oric-Calc is a spreadsheet program which means in quite simple terms that you can enter tables of figures on the screen and then manipulate them in various ways to see how they affect each other. It is particularly useful for budgeting or year-plans where to have to calculate all the figures by hand would take hours. With Oric-Calc you can put in, say how many pairs of tweezers you have in stock and the monthly demand for tweezers and then by entering the appropriate formula work out how many tweezers you would have to buy in for the next ten years, to keep up with demand. Oric-Calc allows you to move sets of figures from one part of the screen to another, total up columns or rows and enter text commentary. As with Oric-Base you are supplied with a

comprehensive manual – this time with worked examples!

Dr. Paul Johnson dropped into my office the other day brandishing his latest acquisition, Zorgons Revenge from I.J.K. Software. This is an animated graphic adventure in four parts (there may be more parts but I couldn't get any further!) – Basically your task is to build a bridge to cross a moat so that you can get into a castle to rescue the princess. Of course there are spiders, carnivorous trees, dinosaurs and strange flapping things who will attempt to stop you. Although technically the graphics and animation are probably the most advanced yet on the Oric it seems to lack the challenge and excitement of I.J.K.s earlier Xenon. I would say that for the person who's tired of space shoot out games, Zorgons Revenge is a refreshing alternative.

Finally I must just mention a game that is close to my heart –



Rat Splat. John Marshall, Geoff Phillips and myself sat round my desk in late July and worked out this game from scratch. The idea was to create the most ridiculous game we could think of. Rat Splat certainly fits the bill. In the sewers live thousands of rats who adore cheese. Your task, as the Sewerman, is to hit the rats with your hammer to stop them eating the cheese you have previously laid as bait. There is a rather nasty monster who would like to eat you and must be deterred by your can of poison gas. Another hazard is that the cheese gives off fumes which if allowed to build up will overpower you. Your smellometer informs you of the increasing danger.

Rat Splat is deliciously gruesome and is a definite must for all psychopathic axe murderers – I love it! I love it!

Next issue, due to some unexpected delays, I hope to look at the Oric Hobbit. So bye for now.

ORIC CONSOLE



An Ideal Home for your Microcomputer

- ★ No More Loose Wires
- ★ Neat Compact Unit
- ★ Supports T.V.
- ★ Cassette Shelf for Software
- ★ Robust Unit
- ★ Protects Computer System
- ★ Improved Safety
- ★ Pad and Pencil Recess

only
£29.95
including VAT

Trade Enquiries Welcome

- FROM YOUR ORIC RETAILER
- OR £29.95 + P+P DIRECT FROM

**EXPRESS COMPUTERS
(RUGBY) LIMITED**

83 HIGH STREET HILLMORTON RUGBY WARWICKSHIRE CV21 4HD TEL: (0788) 71313

Simple Disassembler ...Simpdis

By D. Sinfield

How it came about

That there is a multitude of useful routines concealed in the ORIC's ROM is obvious from the snippets being gradually revealed by this publication. My own interest lies in the addressing of the I/O part of the machine and I was particularly interested in addressing the VIA as although it is already over-worked it could be asked to take on a few more duties.

Simpdis was written to look into the ROM and produce various answers. It is neither quick nor very elegant but it does work!.

How it works

Lines from 10 to 180 set up a set of data strings Line 190 is a patch to call up the short menu routine. Lines 200 to 560 is the disassembler part that builds up a set of output strings:

OUT\$ hold the address, mnemonic and, if applicable, an operand in hex.

LI\$ is the characters represented by the values of the bytes in the instructions. They only print if they are ASCII alphanumerics.

(This should help if you want to convert the program for use with a printer).

Lines 1000 to 1220 put together OUT\$ depending on the addressing mode.

Lines 5000 to 5640 is the data for the data strings: MO\$, BY\$ and OP\$.

Lines 6000 to 6020 Put up a menu and process the choice.

Lines 6100 to 6190 Calculate an offset.

Lines 6200 to 6240 PEEK a byte and tell us all about it.

Lines 6300 to 6320 Call a routine from the program.

Lines 7000 onward print a header while the strings are being loaded.

(1) DIS calls up the disassembler. In answer to the prompts a start and finish address should be given and then the disassembly listing will be produced. Hex or decimal numbers can be used.

The listing can be stopped by pressing any key and then terminated by pressing 'T'.

If the program encounters a number it thinks should be an opcode but it isn't in the 6502 set an ERR message will be produced but the listing will not be interrupted.

A typical line produced would be:

#C000	JMP EA59
↑	↑
Address	Instruction

Ly
↑

This is the ASCII equivalent of 4C and 59. EA is out of the ASCII range and is suppressed.

How to use it

On RUNning the program a choice of functions is presented

Software

#4CEA59



This is the hex equivalent of the instruction.

The third column is included to help sort out the character strings stored.

(2) OFFSET This allows for the calculation of a destination address given the branch address and offset.

(3) PEEK looks at an address & tells you all about it i.e.

Hex location Dec location Dec contents Hex contents and the character if in the range 32-126.

(4) CALL should be used with caution. It can leave you looping without any hope of an elegant escape and CALLing EA59 or C000 will do a cold start and destroy any program you may have in.

Simpdis Listing

```

10 REM *****
20 REM          SIMPDIS
30 REM          SIMPLE DISASSEMBLER
40 REM
50 REM *****
60 REM *****
65 GOSUB7000:POKE618,2
70 DIMBP$(255),BY$(255),MD$(255)
100 FORA=0TO255
110 READOP$(A)
120 NEXT
130 FORA=0TO255
140 READBY$(A)
150 NEXT
160 FORA=0TO255
170 READMD$(A)
180 NEXT
190 GOTD6000
200 INPUT"START LOCATION";S$
210 S=VAL(S$)
220 INPUT"FINISH LOCATION";F$
230 F=VAL(F$)
240 IFS>=FTHEN200
250 REPEAT
260 N=PEEK(S)
270 OUT$=HEX$(S)+" "+LEFT$(OP$(N),3)
272 LI$="":X$=""
275 IFBY$(N)="1"THEN520
280 IF BY$(N)="2"THENX$=HEX$(PEEK(S+1)):GOTO500
290 X$=HEX$(DEEK(S+1))
500 B=LEN(X$)-1:X$=RIGHT$(X$,B)
510 ON VAL(MD$(N)) GOSUB1000,1050,1150,1200,1200
520 FORA=1TOVAL(BY$(N))+.5

```

```

522 Q=PEEK(S+A-1)
525 IFQ<320RQ>126THEN532
530 LI$=LI$+CHR$(Q)
532 NEXT
540 PRINTOUT$;:POKE#269,20:PRINTLI$;:POKE#269,26:PRINTHEX$(N);X$
550 S=S+VAL(BY$(N))
551 RP$=""
555 IF KEY$<>" "THENGETRP$
557 IFRP$="T"THENPULL:GOTO6000
560 UNTILS>F
570 GOTO6000
1000 OUT$=OUT$+" "+X$
1010 RETURN
1020 PRINTHEX$(N),OP$(N),BY$(N),MD$(N)
1030 NEXT
1050 IFRIGHT$(OP$(N),1)="X"THEN1090
1060 OUT$=OUT$+" (" +X$+" ),Y"
1070 RETURN
1090 OUT$=OUT$+" (" +X$+" ,X)"
1100 RETURN
1150 OUT$=OUT$+" #"+X$
1160 RETURN
1200 IF LEN(OP$(N))=3THEN1000
1210 OUT$=OUT$+" "+X$+" ,"+RIGHT$(OP$(N),1)
1220 RETURN
5000 DATABRK,ORAX,ERR,ERR,ERR,ORA,ASL,ERR
5010 DATAPHF,ORA,ASL,ERR,ERR,ORA,ASL,ERR,BPL,ORAY
5020 DATA ERR,ERR,ERR,ERR,ORAX,ASLX,ERR,CLC,ORAY
5030 DATAERR,ERR,ERR,ORAX,ASLX,ERR,JSR
5040 DATAANDX,ERR,ERR,BIT,AND,ROL,ERR
5050 DATAPLP,AND,ROL,ERR,BIT,AND
4962 DATAROL,ERR,BMI,ANDY,ERR,ERR,ERR
5070 DATAANDX,ROLX,ERR,SEC,ANDY
5080 DATAERR,ERR,ERR,ANDX,ROLX,ERR
5090 DATARTI,EORX,ERR,ERR,ERR,EOR,LSR
5100 DATA ERR,PHA,EOR,LSR,ERR,JMP,EOR,LSR,ERR
5110 DATABVC,EORY,ERR,ERR,ERR,EORX,LSRX
5120 DATAERR,CLI,EORY,ERR,ERR,ERR,EORX
5130 DATALSRX,ERR,RTS,ADCX,ERR,ERR,ERR,ADC
5140 DATAROR,ERR,PLA,ADC,ROR,ERR,JMP,ADC
5151 DATAROR,ERR,BVS,ADCY,ERR,ERR,ERR,ADCX
5160 DATARORX,ERR,SEI,ADCY,ERR,ERR,ERR,ADCX
5170 DATARORX,ERR,ERR,STAX,ERR,ERR,STY,STA
5180 DATASTX,ERR,DEY,ERR,TXA,ERR,STY,STA
5190 DATASTX,ERR,BCC,STAY,ERR,ERR,STYX,STAX
5200 DATASTXY,ERR,TYA,STAY,TXS,ERR,ERR,STAX
5210 DATAERR,ERR,LDY,LDAX,LDX,ERR,LDY,LDA
5220 DATALDX,ERR,TAY,LDA,TAX,ERR,LDY,LDA
5230 DATALDX,ERR,BCS,LDAY,ERR,ERR,LDYX
5240 DATALDX,LDXY,ERR,CLV,LDAY,TSX,ERR,LDY
5250 DATA LDAX,LDXY,ERR,CPY,CMPX,ERR,ERR,CPY
5260 DATACMP,DEC,ERR,INY,CMP,DEX,ERR,CPY,CMP,DEC
5270 DATAERR,BNE,CMP,ERR,ERR,ERR,DEC,ERR,ERR
5280 DATACLD,CPY,ERR,ERR,ERR,CMPX,DEC,ERR,CPX
5290 DATASBCX,ERR,ERR,CPX,SBC,INC,ERR,INX,SBC,NOP
5300 DATAERR,CPX,SBC,INC,ERR,BEQ,SBCY,ERR,ERR
5310 DATAERR,SBCX,INX,ERR,SED,SBCY,ERR,ERR
5320 DATAERR,SBCX,INX,ERR

```

```

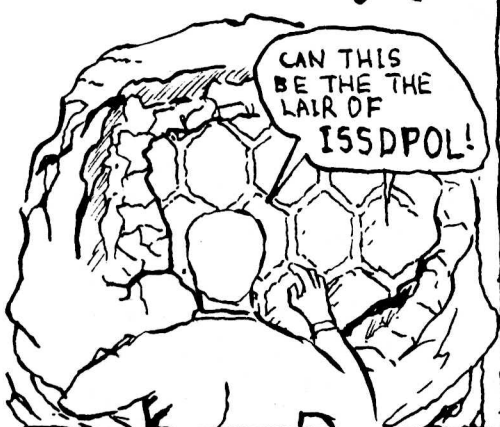
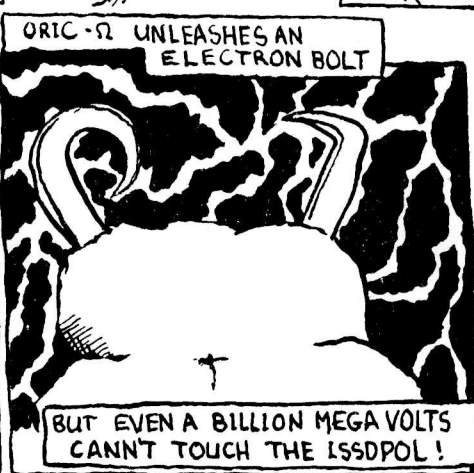
5330 DATA1,2,1,1,1,2,2,1,1,2,1,1,1,3,3,1
5341 DATA2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
5350 DATA3,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1
5360 DATA2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
5370 DATA1,2,1,1,1,2,2,1,1,2,3,1,3,3,3,1
5380 DATA2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
5390 DATA1,2,1,1,1,2,2,1,1,2,1,1,3,3,3,1
5400 DATA2,2,1,1,1,2,2,1,1,3,1,1,1,2,2,1
5410 DATA1,2,1,1,2,2,2,1,1,1,1,1,3,3,3,1
5420 DATA2,2,1,1,2,2,2,1,1,3,1,1,1,3,1,1
5430 DATA2,2,2,1,2,2,2,1,1,2,1,1,3,3,3,1
5440 DATA2,2,1,1,2,2,2,1,1,3,1,1,3,3,3,1
5450 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1
5460 DATA2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
5470 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1
5480 DATA2,2,1,1,1,2,2,1,1,2,1,1,1,3,3,1
5485 REM MODE
5490 DATA1,2,1,1,1,4,4,1,1,3,1,1,1,1,1,1
5500 DATA1,2,1,1,1,4,4,1,1,1,1,1,1,1,1,1
5510 DATA1,2,1,1,4,4,4,1,1,3,1,1,1,1,1,1
5520 DATA1,2,1,1,1,4,4,1,1,5,1,1,1,5,5,1
5530 DATA1,2,1,1,1,4,4,1,1,3,1,1,1,1,1,1
5540 DATA1,2,1,1,1,4,4,1,1,5,1,1,1,5,5,1
5550 DATA1,2,1,1,1,4,4,1,1,3,1,1,2,1,1,1
5560 DATA1,2,1,1,1,4,4,1,1,5,1,1,1,5,5,1
5570 DATA1,2,1,1,4,4,4,1,1,1,1,1,1,1,1,1
5580 DATA1,2,1,1,4,4,4,1,1,5,1,1,1,5,1,1
5590 DATA3,2,3,1,4,4,4,1,1,3,1,3,1,1,1,1
5600 DATA1,2,1,1,4,4,4,1,1,5,1,1,5,5,5,1
5610 DATA3,2,1,1,4,4,4,1,1,3,1,1,1,1,1,1
5620 DATA1,2,1,1,1,4,4,1,1,5,1,1,1,5,5,1
5630 DATA3,2,1,1,4,4,4,1,1,3,1,1,1,1,1,1
5640 DATA1,2,1,1,1,4,4,1,1,5,1,1,1,5,5,1
6000 PRINT"1=DIS:2=OFFSET:3=PEEK:4=CALL"
6010 GETJ
6020 ON J GOTO 0200,6100,6200,6300
6100 INPUT"HEX OFFSET";0
6110 INPUT"FROM ADDRESS";F
6120 IFD>#7FTHEN6170
6130 D=F+0+2
6140 PRINT"DESTINATION=";HEX$(D)
6150 GOTO6000
6170 O=#FE-0
6180 D=F-0
6190 GOTO6140
6200 INPUT"BYTE";B
6210 N=PEEK(B)
6220 PRINTHEX$(B),B,N,HEX$(N)
6230 IFN<320RN>127THENPRINT:GOTO6000
6240 PRINTCHR$(N):GOTO6000
6300 INPUT"ROUTINE START";N
6310 CALLN
6320 GOTO6000
7000 CLS:PRINT:PRINT:PRINT:PRINT:PRINT
7010 PRINT"*****"
7020 PRINT:PRINT"SIMPLE DISASSEMBLER"
7030 PRINT:PRINTSPC(5);CHR$(96);"DAVID SINFIELD"
7050 PRINT:PRINT"TEL. (0485 70865)"
7060 PRINT:PRINT"*****"
7070 RETURN

```

Simpodis Listing

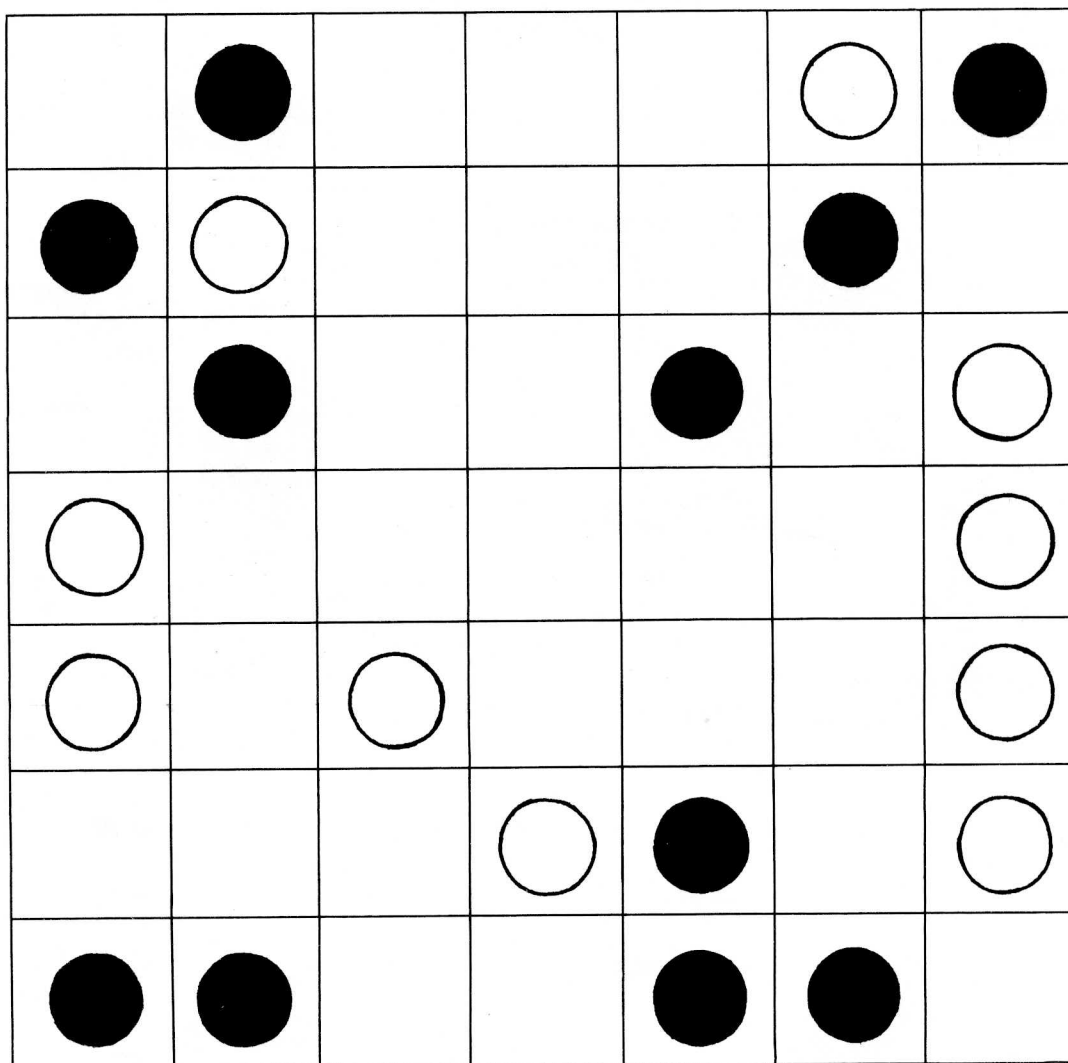
Software

CAPTAIN TANEX



'Connect-4'

By T. M. Mughal



'CONNECT-4' is a 2-player game in which the two players drop their counters into a vertical grid and attempt to get '4-in-a-row' (vertically, horizontally or diagonally). This program takes the place of the grid and counters and makes good use of sound, colour and high resolution graphics. The program may be re-run in two ways in the middle of a game:

1. by pressing 'N': this keeps the names of the current two plays and merely redraws the board (i.e. empties the grid). This key ('N') should be pressed when someone has won.
2. by pressing 'R': this runs the program all over again. This key should be pressed when different players want to play

(as new names are requested).

The program is very easy to use and very 'user-friendly'. It rejects erroneous inputs and automatically clears the board and starts again when the grid has become full and no-one has requested a new game or a re-run of the program.

'Connect-4' Listing

```

10 HIRES:POKE226A,10
20 DEF FNR(X)=INT(RND(1)*X+1):GOSUB30:GOTO160
30 PRINTFRE("):CLS:FORX=10TO220STEP30
40 CURSET X,10,1
50 DRAW 0,180,1
60 NEXT X
70 FOR Y=10 TO 190 STEP 30
80 CURSET 10,Y,3
90 DRAW 210,0,1
100 NEXT Y:I=0
110 FOR X=25 TO 205 STEP 30
120 CURSET X,191,3
130 I=I+1
140 CHAR 48+I,0,1
150 NEXT:M=0:GOSUB2500:RETURN
160 PRINTCHR$(12)
170 PRINTSPC(11);CHR$(4);CHR$(27);"J";CHR$(27);"VConnect 4 ";CHR$(27);"W";
180 PRINTCHR$(4)
190 REM
200 WAIT 300:PRINTCHR$(12)
210 PRINTSPC(7);CHR$(4);CHR$(27);"J"CHR$(96)" Tarik Mughal 1983"CHR$(4)
220 REM
230 WAIT 300:PRINTCHR$(12):POKE226A,11
240 INPUT"Enter name of player on left";P1$:PRINTCHR$(12)
250 INPUT"Enter name of player on right";P2$:CLS:GOSUB3000
260 PRINTP1$;" playing red."
270 PRINTP2$;" playing yellow."
280 PRINT" CHR$(27)"D"CHR$(27)"VPress any key... "CHR$(27)"W";
290 GETZ$:CLS:POKE226A,10
300 REM
310 REM
320 REM
330 REM
340 FORI=1 TO 7:H(I)=0:NEXT I
350 IF FNR(2)=1 THEN 410
360 PL=0
370 PRINTP1$;"'s go..."
380 PRINTQ$;
390 GOSUB 1000
400 GOSUB 2000
410 PL=1
420 PRINT P2$;"'s go..."
430 PRINTQ$;
440 GOSUB 1000
450 GOSUB 2000

```

```

460 GOTO 360
1000 GET Z$:CLS
1010 C=VAL(Z$)
1020 IF Z$="N" THEN 1200
1025 IF Z$="R" THEN 1400
1030 REM
1040 IF C>0 AND C<8 AND H(C)<6 THEN RETURN
1045 IF H(C)>5 THEN PRINTCHR$(12)"That column is full!! - try again
":GOTO 1000
1050 PRINT"Out of range!!! - try again"
1060 GOTO 1000
1200 PRINT"New game ? Are you sure (Y/N) ?"
1210 GET Z$:IF Z$="Y"THENHIRES:GOSUB30:GOTO260
1220 IF Z$<>"N"THEN1210
1230 PRINT"O.K. , enter column...":GOTO1000
1400 PRINT"Re-run ? Are you sure ?"
1410 GETZ$:IF Z$="Y"THENRUN
1420 IF Z$<>"N"THEN1410
1430 GOTO1230
2000 IF PL=0 THEN CL=1 ELSE CL=3
2010 X=12+(C-1)*30
2020 Y=161-H(C)*30
2030 CURSET X,Y,3
2040 FILL 29,1,CL
2050 X=35+(C-1)*30
2060 CURSET X,Y,3
2070 FILL 29,1,7
2080 X=25+(C-1)*30
2090 Y=175-H(C)*30
2100 CURSET X,Y,3:PLAY1,0,1,1
2110 MUSIC1,CL,5,2:FORR=1TO5
2120 CIRCLES,1
2130 NEXT R:PLAY0,0,0,0
2140 H(C)=H(C)+1:M=M+1
2150 IF M<42 THEN RETURN
2160 PLAY1,0,0,0:FORV=15TO1STEP-1:FORF=40TO250STEP15:Z$=KEY$
2170 SOUND1,F,V:NEXTF,V:PLAY0,0,0,0
2180 PRINTCHR$(12)"The board is full !!!"
2190 WAIT200:PRINTCHR$(12)"Get ready for a new game !":WAIT200
2200 PRINTFRE("):HIRES:GOSUB30:GOTO260
2500 FORY=2A000TO2BDB0STEP40
2510 POKEY,6:NEXT
2520 Q$=" "+CHR$(27)+"L"+CHR$(27)+"G"+CHR$(27)+"P"+"Enter co
lumn..."
2525 Q$=Q$+CHR$(27)+"W"+CHR$(10)+CHR$(13)
2530 Q$=Q$+"('R' to re-run , 'N' for a new game)"
2540 RETURN
3000 P1$=" "+CHR$(27)+"G"+CHR$(27)+"P"+P1$+CHR$(27)+"W"+CHR$(27)+"@
"
3010 P2$=" "+CHR$(27)+"G"+CHR$(27)+"P"+P2$+CHR$(27)+"W"+CHR$(27)+"@
"
3020 RETURN

```

Oric-Base revealed



By G. M. Phillips

Program Errors

First of all, there are two REAL errors which can be corrected.

The first is that SUBTRACT may produce the wrong sign – 2 from 5 giving –3, for example.

To correct this, load up Oricbase, press CONTROL and C until the message READY appears, and type:

```
8510 O9$=STR$(O4-
```

```
O3):GOTO8690
```

thus swapping around the 03 and 04.

Whilst you are doing this, you may as well correct the minor fault whereby DUMP can print fields that are slightly out of place, after calculations have been done. Type in:
8711 IF ASC(O9\$)=2 THEN
O9\$=MID\$(O9\$,2)

This fault is due to one of the bugs in Oric Basic.

Loading problems

If you cannot load at fast speed, I would recommend that you load up at slow, and then CSAVE at fast either back on to the Oricbase cassette or onto another cassette. You will almost certainly have more luck with your own recording at fast – and if not then you cannot blame Oricbase for that!

File Handling

If you are using the RELAY connections on the cassette recorder, then loading and saving files of data should be successful, but if not then read on: If you have a file of reasonable size, Oricbase may 'miss' your macro file, and perpetually wait for it on the tape.

One solution, if your tape does not have a relay feature is to rewind the tape to the second file if Oricbase misses it the first time around.

Using the Oricbase Program

This is the first in a series of articles looking at the Oric-base database program.

In this issue, I will present a couple of 'fixes' for those of you who are experiencing problems in using the program.

By the time you read this article, new Oricbase programs will have been corrected – so check your program before altering it.

Corrected programs will show up as 'ORICBASE V.2'.

Features



A better solution – which will be implemented on future versions of this program, so check to see if this is a problem first – is to change a couple of lines in the program. This will not take you more than a minute:

```
1077 POKE#67,O:CALL#  
B85B,D$  
1130 POKE#67,O:CALL#  
B85B,M$
```

The fault here lies with the 'standard' cassette file routines which have the irritating ten-

dency to call Garbage collection.

Once you have made this correction, both data and macros should load with no difficulty. If not, then try cleaning your tape heads. Also beware of putting impossible sized files on Oribase.

These four lines are all you need to change – once you have done this, resave the program back onto another cassette (or onto the fast side of the Oribase tape).

The 'FIND' command

One of the most common misunderstandings about Oribase is in conjunction with the FIND command.

The FIND can only compare fields against actual values, numeric or string, and not against registers (e.g. #1) or other fields (e.g. NAME).

For example, if a file contains fieldnames: NAME, AGE, PHONE, then FIND NAME < PHONE is INVALID. To Oribase, you are asking for any names which have a value less than 'PHONE' – such as JONES, MALCOLM, but not PHOTO or SMITH.

If you want to do this kind of thing, you would have to use a

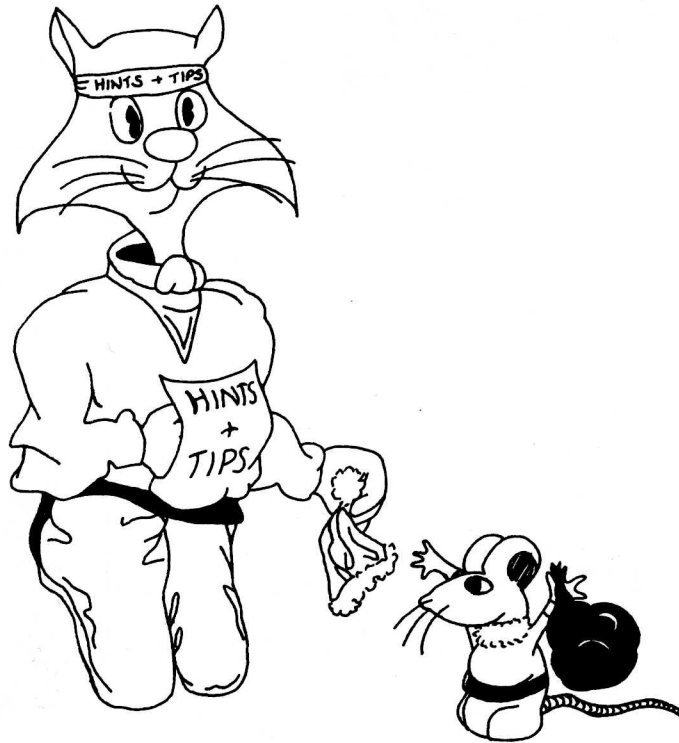
IND that encompasses your whole file, followed by an IF statement after the ATR command.

```
e.g. FIND NAME > " " ATR  
IF VALUE1 < VALUE2  
PRINT NAME ENDIF
```

The next article will discuss some of the ways that you can use FIND in more detail than the manual. So dust off your Oribase cassettes, correct those four lines, if necessary, and get Oribasing!

Hints & Tips

By T. M. Mughal



Flashing Rems

This short routine provides a flashing REM's facility. If you want a REM statement to flash in your program listing just put TWO spaces immediately after the REM.

i.e. <line-no> REM <space><space><comment>

Then when you have written your program type: GOSUB 60000. Then list your program. All your REM statements which had two spaces after the REM will be flashing.

(If you don't want a REM to flash then don't put two spaces between the REM and the comment).

Computed Restore Routine

The ORIC-1's RESTORE facility does not allow the flexibility of RESTORE'ing the data pointer to a particular line numbers. i.e. RESTORE <line-no> is not allowed.

The ORIC-1 RESTORE just sets the data pointer to the first data item in the program.

The COMPUTED RESTORE ROUTINE allows the equivalent of RESTORE <line-no>. To use it, type:

```
XX = <line-no>: GOSUB  
60000
```

where <line-no> is the line number you want the data pointer set to.

The routine is fairly well mugged:

If you set XX to a non-existent line number, it will RETURN without executing.

If you set XX to a line number which does not have DATA as its first statement, the routine will RETURN without executing.

All this in just seven lines!

Hints & Tips

By B. Edel



The first routine (Listing 1) returns the row and column values of the cursor place after positioning it with the four cursor control keys from keyboard. The values are affected continuously to the variables ROW and COL until the RETURN key is pressed. The program is written so that the cursor cannot pass the upper or lower limit of the screen board.

The consequence is that the scrollins of the screen is not possible.

If the current position of the cursor is unknown, call the routine at line 10000. The cursor will so home before the variables are initialized with the corre-

sponding values (i.e. ROW=1 and COL=-1). If not, skip the input lines and call at line 10030. After manipulation with the cursor control keys, press the RETURN key to end the routine. All other keys will have no effect. The values of the variables are showed in the screen last row. If the display is not wanted, the PLOT statements can be removed in the following lines of the program: 10220, 10230 and 10240.

If the program is called with a GOSUB, replace the END at line 10260 with a RETURN statement.

The other routine (Listing 2) allows the user to send the cur-

sor to a predetermined row and column. If the values of row and column must be entered through the keyboard, then call the routine at line 20000. If these values are computed in the main program, affect this to the variables ROW and COL respectively and call at line 20030.

If the routine is called with a GOSUB, replace the END in line 20130 with RETURN.

CAUTION: In accordance with the ORIC manual, the values of the rows are 0 to 26, and the values of columns are -1 to 38.

Hints & Tips Listings

FLASHING REMS

```
60000 L=1281
60010 NXT=DEEK(L):IF NXT=Ø THEN RETURN
60020 IF PEEK(L+4)=157 AND DEEK(L+5)=8224 THEN 60040
60030 L=NXT:GOTO 60010
60040 DOKE L+5,19483:GOTO 60030
```

COMPUTED RESTORE ROUTINE

=====

```
60000 L=1281
60010 NXT=DEEK(L):IF NXT=Ø THEN RETURN
60020 IF DEEK(L+2)=XX THEN 60050
60030 IF DEEK(L+2)>XX THEN RETURN
60040 L=NXT:GOTO 60010
60050 IF PEEK(L+4)=145 THEN DOKE 176,L-1
60060 RETURN
```

```
10000 REM * RETURN CURSOR-POSITION *
10010 PRINT CHR$(30)
10020 LET ROW=1 : LET COL=-1
10030 GET W$
10040 IF W$=CHR$(13) THEN 10260
10050 IF W$=CHR$(11) THEN 10100
10060 IF W$=CHR$(10) THEN 10120
10070 IF W$=CHR$(9) THEN 10140
10080 IF W$=CHR$(8) THEN 10180
10090 GOTO 10030
10100 LET ROW=ROW-1 : IF ROW>-1 THEN 10220
10110 LET ROW=0 : GOTO 10030
10120 LET ROW=ROW+1 : IF ROW<27 THEN 10220
10130 LET ROW=26 : GOTO 10030
10140 LET COL=COL+1
10150 IF COL<39 THEN 10220
10160 IF ROW=26 THEN COL=38 ELSE COL=-1 : GOTO 10120
10170 GOTO 10030
10180 LET COL=COL-1
10190 IF COL>-2 THEN 10220
10200 IF ROW=0 THEN COL=-1 ELSE COL=38 : GOTO 10100
10210 GOTO 10030
10220 PRINT W$:: PLOT 1,26,CHR$(2)
10230 PLOT 20,26,"COL="+STR$(COL)+" "
10240 PLOT 10,26,"ROW="+STR$(ROW)+" "
10250 GOTO 10030
10260 END
```

Listing 1

```
20000 REM * CURSOR POSITIONING *
20010 INPUT "ROW CURSOR WANTED":ROW
20020 INPUT "COL CURSOR WANTED":COL
20030 PRINT CHR$(30)
20040 IF COL<-1 OR COL>38 THEN PRINT "COL.ERROR" : GOTO 20130
20050 IF COL=-1 THEN 20080
20060 FOR I=0 TO COL
20070 PRINT CHR$(9):: NEXT I
20080 IF ROW<0 OR ROW>26 THEN PRINT "ROW ERROR" : GOTO 20130
20090 IF ROW=0 THEN PRINT CHR$(11):: GOTO 20130
20100 IF ROW=1 THEN 20130
20110 FOR I=2 TO ROW
20120 PRINT CHR$(10):: NEXT I
20130 WAIT 100 : END
```

Listing 2

Hints & Tips

Composite Video From The Oric 1 Microcomputer

By I. McLean

The Oric 1 provides an output to an RGB colour monitor. There are occasions, however, when a composite video signal may be required.

I needed this signal, for example, to take screen dumps using an electrothermal printer. It would also be required if the micro were to be linked to a low cost black and white monitor.

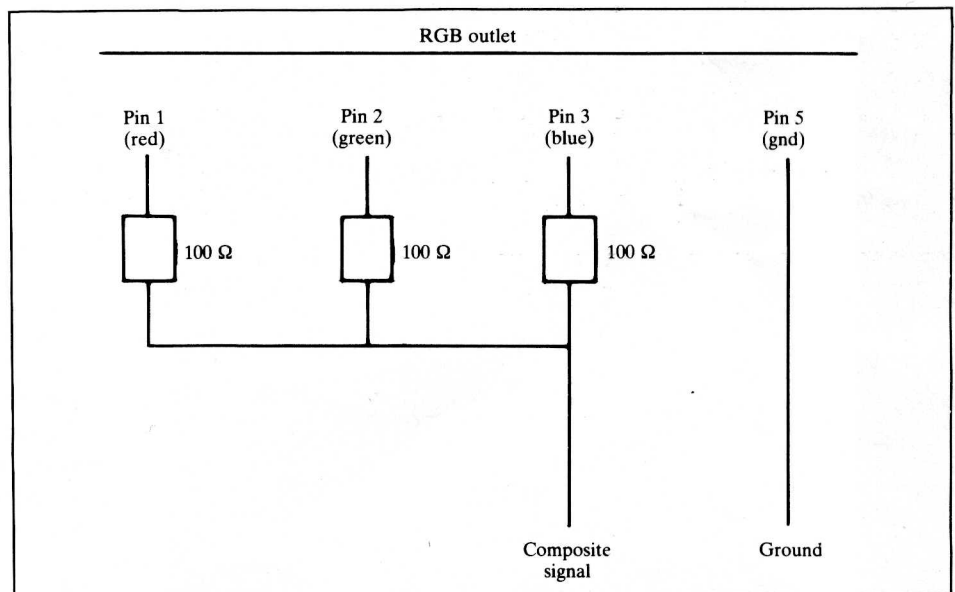
One method of getting this signal is to open up the Oric and take a line out from the input to the TV signal modulator. This has three major disadvantages:

1. It invalidates the guarantee.
2. You have to work out exactly where to solder on your wire – and can damage the machine while doing so.
3. The signal thus obtained is very weak.

A better method is to make up a connector which takes the Red, Green and Blue signals from the RGB outlet and mixes them to form a composite signal. This

sounds more complicated than it is. A simple resistor mixer is all that is required.

The circuit is:



The components required are three 100 ohm resistors (1/4 watt, 10% tolerance types are adequate), a 5 pin DIN plug and a co-axial video input plug (similar to a TV input plug). It may be necessary to cannibalise two cable connectors to get the plugs, but these are cheap and readily obtainable.

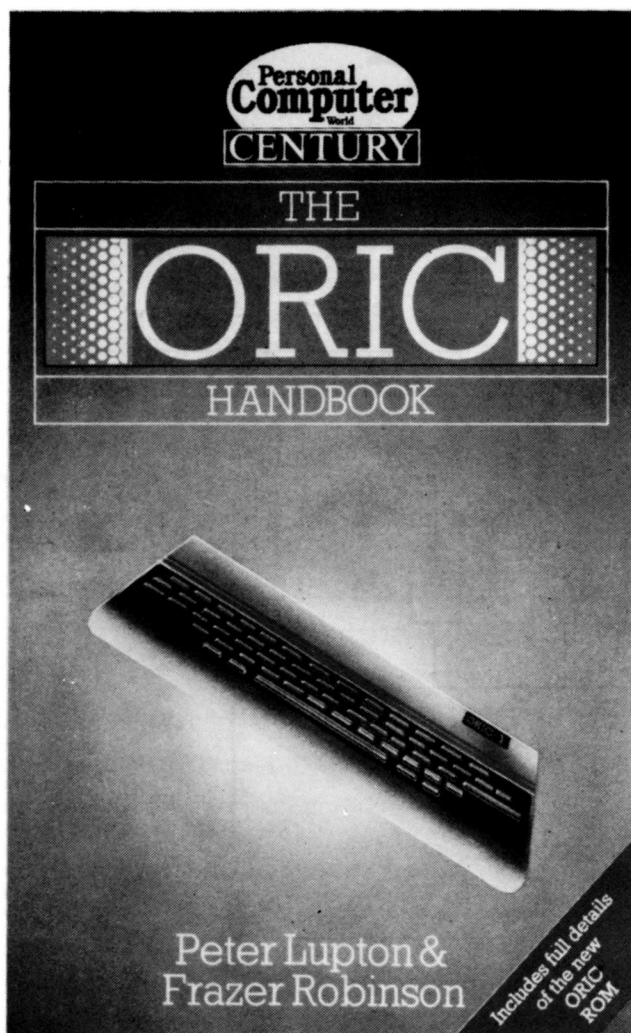
The physical locations of pins 1 to 5 are normally marked on the

DIN plug. If not, page 151 of the User Manual gives the necessary information.

The signal generated will not give colour, although it could be adapted to do so. If colour is important to you then stick with an RGB monitor or a colour television.

TAKE YOUR ORIC TO THE LIMITS -AND BEYOND

The Oric Handbook



PETER LUPTON and FRAZER ROBINSON

The Oric Handbook offers its readers the opportunity of harnessing the power of one of the newest and most exciting microcomputers. A clear step-by-step introduction opens the Oric to the beginner while the wealth of hints and tips, exciting programs and applications makes the book essential reading for even the experienced programmer. There are also lengthy sections on sound and graphic capabilities which are particularly impressive features of the Oric.

The book contains full details of the differences between the old and new ORIC ROM and also documents all the new features and commands.

224pp **£5.95**

*Available through your local bookshop
or if you experience any difficulty
please fill in the form below*

ORDER FORM

To: George Philip Services Ltd,
Arndale Road, Wick, Littlehampton,
West Sussex BN17 7EN

Please send me _____ copy/copies of
THE ORIC HANDBOOK by **PETER LUPTON** and
FRAZER ROBINSON at £6.50 per copy (post paid)

I enclose my cheque/postal order for £6.50 per
copy
Please make payable to George Philip

Name _____

Address _____

Please allow up to 28 days for delivery

Features

We will be featuring extracts from the Oric Handbook (pictured opposite), in this and the coming issues of Oric Owner. This time we are featuring Chapters 3 and 4 on Communication & Programming.

CHAPTER 3 COMMUNICATION

Before you can write programs for your ORIC, you must find out how to communicate with it.

Communication is a two-way process: you must give the ORIC instructions, and you must be able to find out how it responds to them. You give instructions using the keyboard, and the ORIC displays its response on the TV screen.

Type the following phrase on the keyboard.

PRINT "ORIC-1"

You will see the letters appear on the screen as you type them, and hear a 'click' as you press each key. The flashing square – called the cursor – will move to indicate where the next letter you type will appear. Nothing else happens though – the ORIC has not yet obeyed your instruction.

Now press the RETURN key. The words 'ORIC-1' appear on the screen, and the word 'Ready' is printed to tell you that the ORIC is waiting for your next command. (If instead of printing 'ORIC-1' the computer prints '?SYNTAX ERROR' or 0, it means you have made a typing mistake. Try again!)

So, to give the ORIC a command you type it at the keyboard and press RETURN. Try another command:

PRINT "HELLO!"

Again, the letters between the quotation marks are printed. You can tell the ORIC to print any sequence of letters, but you must remember to put them between quotation marks. Try making it print your name.

You don't have to type PRINT in full everytime – the question mark has the same meaning to the ORIC. Try

? "HELLO!"

(remembering to press RETURN, of course).

MANAGING THE DISPLAY

As well as typing on to the screen, there are a number of different ways to alter the display.

The Cursor

This can be moved around the screen using the keys marked with arrows at either end of the space bar. Try moving the cursor around, and watch what happens when it gets to the edge of the picture: when it reaches one side it reappears at the other side, but when the cursor reaches the top or bottom of the screen the picture starts to move in the opposite direction. This vertical movement is called **scrolling**. Notice that anything that disappears at the top or bottom when the picture scrolls cannot be recovered by scrolling the other way: it is lost for ever.

Corrections

If you notice that you have made a typing mistake before you press RETURN, you can correct the error by using the DEL key. For example, if you type

PRONT "ORIC"

and then realise your mistake, press DEL until the first O is erased and then retype the rest of the line.

There is another way of making corrections, or editing what you have typed, which is more convenient for major corrections than retyping.

NOTE You can **not** make corrections by moving the cursor over the error with the arrowed keys and then retyping. The ORIC will not understand what you have done and will display the '?SYNTAX ERROR' message.

Clearing the screen

This can be done in two ways. You can either hold down the CTRL key and press L, or type CLS and press RETURN.

Features

NUMBERS

As well as printing words, the ORIC can also handle numbers. Try

PRINT 5 (RETURN)

You can do arithmetic.

Addition:

PRINT 5+3

Subtraction:

PRINT 7-4

Multiplication:

PRINT 3*5

Division:

PRINT 15/6

Powers(exponentiation):

PRINT 3 ↑ 2

You can ask the ORIC to calculate longer expressions, such as:

PRINT 3*5 + (5-3)/(2+1) * 7/8

In working out expressions like this, the ORIC follows strict rules about the order in which the various arithmetical operations are performed.

First The expressions inside brackets are evaluated.

Second Any powers (squares, cubes, etc.) indicated by ↑ are worked out.

Third The ORIC performs the multiplications and divisions.

Fourth The additions and subtractions are performed.

If you are unsure, put brackets round the bit you want calculated first. Try the following examples, and see if you can work out the answers yourself to check that you know what the ORIC is doing.

PRINT 2 * 2 + 2 * 2

PRINT 2 + 2 * 2 + 2

PRINT 6 + 3 / 5 - 4

PRINT 3 ↑ 2 + 4 ↑ 3/2

PRINT (3 + 4) * 2

COLOUR

The colours of the writing on the screen and of the background can be changed. The foreground colour (the colour of the writing) is altered by the command **INK** and the number of the desired colour. Try this:

INK 1

The writing on the screen will turn red.

The background is changed in a similar way by the **PAPER** command.

PAPER 3

will turn the background yellow.

There are eight colours, and their numbers are as follows:

BLACK	0
RED	1
GREEN	2
YELLOW	3
BLUE	4
MAGENTA	5
CYAN	6
WHITE	7

Try changing the foreground and background colours to test the possibilities. If all the writing on the screen disappears don't panic; you will have set the **INK** and **PAPER** to the same colour. The ORIC will still understand commands, even if you can't see what you are typing.

SOUNDS

There are four special sounds which the ORIC will make in response to your commands. These are:

PING
SHOOT
ZAP
EXPLODE

To hear a sound, type the name and press RETURN.

MULTIPLE COMMANDS

You can put more than one instruction in one command if you separate the instructions with colons(:). Try this:

Features

CLS: PRINT "3 + 4 = "3+4: PING

Or this:

CLS: PAPER 3: INK 4: ZAP: PRINT: PRINT
"BANG!": EXPLODE

It doesn't matter if you run over the end of the line on the screen, as long as there are no more than 78 characters altogether in the command (including the spaces). The ORIC will warn you by making the 'PING' sound when you get near the limit.

Entering a number of instructions together like this can get very cumbersome. In the next chapter we will discover a way of giving the computer a very large number of instructions all at once – the **PROGRAM**.

SUMMARY

PRINT tells the ORIC to print something.

PRINT "ABCDEFGH"

prints the characters between the quotation marks.

PRINT 3+4

prints the result of the sum.

These can be combined:

PRINT "3+4*5+7 = " 3+4*5+7
SOUNDS **ZAP**
 SHOOT
 PING
 EXPLODE

COLOURS **INK** changes the foreground,
 PAPER changes the background.

Both **INK** and **PAPER** must be followed by a number between 0 and 7.

MULTIPLE COMMANDS must be separated by colons(:).

CHAPTER 4

PROGRAMMING

At the end of Chapter 3, we saw that it is possible to give the ORIC a number of instructions at one

time by writing them one after another on one line. We will now look at a much more powerful way of doing this – the **program**.

A computer program is a number list of computer instructions which are entered together and stored by the computer to be obeyed later. Let's look at the last example of Chapter 3 and see how we can turn it into a program. In Chapter 3 we had:

CLS: PAPER 3: INK 4: ZAP: PRINT: PRINT
"BANG!": EXPLODE

Rewritten as a program it looks like this:

```
10 CLS
20 PAPER 3
30 INK 4
40 ZAP
50 PRINT
60 PRINT "BANG!"
70 EXPLODE
```

Type in each line in turn (remembering to press RETURN after each). You will see that the ORIC does not obey the instructions as they are entered, and that the 'READY' message does not appear after you press RETURN. The ORIC recognises that each line is part of a program, because each begins with a number, and the program is stored for future use.

When you have typed in all the lines, you can inspect the stored program by typing **LIST** (and pressing RETURN, of course). The instructions are listed in numerical order on the screen. You can list parts of the program by specifying line numbers. For example:

```
LIST 30-50   lists all lines from 30 to 50
LIST 20       lists only line 20
LIST - 30     lists all lines up to and including
               line 30
LIST 30 -     lists all lines from line 30 to the
               end of the program
```

To order the computer to act on the program instructions use the command **RUN** (followed by RETURN). This tells the ORIC to read through the stored program and obey each instruction in order. You will see that the program gives exactly the same results as the multiple instructions in Chapter 3. (If it doesn't, and the ORIC prints a

Features

message such as 'SYNTAX ERROR IN LINE 20', it means there is a typing mistake in that line and the ORIC cannot understand the instruction. **LIST** the line to check, and retype it.)

A program can be **RUN** as many times as you like. It will be stored in the computer's memory until you switch it off, unless you alter or delete the program.

It is possible to begin execution of a program from a line other than the first by specifying the line number. For example **RUN 20** begins the program at line 20, and line 10, the **CLS** instruction, is ignored.

Programs can be altered or added to at will. To alter an instruction, retype the line. If you enter:

```
20 PAPER 1
```

the original line 20 will be replaced by the new one, and the next time you **RUN** the program the screen will turn red instead of yellow. (To restore the black and white screen after running the program you can type

```
PAPER 7: INK 0
```

and press RETURN.)

Extra lines are added by typing them as before:

```
35 ZAP: ZAP: ZAP
```

makes the program even noisier. Notice that the ORIC inserts the extra line in the appropriate place: it is **not** put at the end.

NOTE It is a good idea always to number program lines in steps of 10 to allow extra or forgotten lines to be inserted without having to renumber the whole program.

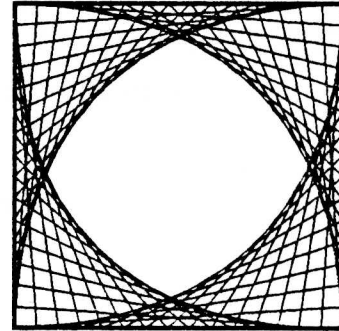
To delete a line, type the line number and press RETURN. The line is deleted from the list in the ORIC's memory.

Now see if you can alter the example program to give different **INK** and **PAPER** colours, and to **PRINT** different words on the screen.

When you have exhausted the possibilities of this program, you can use the command **NEW** to delete the whole program. You will then be ready for the next program.

PRETTY PATTERNS

Here is a more interesting program for you to try. The program draws a pattern like the one shown in the diagram below, using the ORIC's high resolution graphics facilities. Many of the commands used will be unfamiliar to you at present, but you need not worry about that. Just type in the program and watch what it does.



```
10 TEXT
20 PAPER6: INK4
30 HIRES
40 PAPER6: INK4
50 PRINT CHR$(17)

100 CURSET 20,0,1
110 DRAW 200,0,1
120 DRAW 0,199,1
130 DRAW -200,0,1
140 DRAW 0,-199,1

200 FOR X=0 TO 199 STEP 3
210 CURSET 21,X,3
220 DRAW X, 199-X, 1
230 DRAW 199-X, -X, 1
240 DRAW -X, X-199, 1
250 DRAW X-200, X, 1
260 NEXT X

300 PRINT CHR$(17)
310 END
```

Check the program carefully for typing errors, and then **RUN** it. If the program stops and the message '?SYNTAX ERROR IN LINE xxx' is printed, it means you have made a typing mistake in that line. If any other error message is printed, check the whole of the program for mistakes.

To restore the normal display after running the program, type **TEXT** and press RETURN. The screen can not be cleared by **CLS** or CTRL and L

Features

because the pattern is drawn in high resolution mode. This will all be made clear in Chapter 13, so don't worry about it at this stage.

ARITHMETIC IN PROGRAMS – VARIABLES

We discovered in Chapter 3 that the ORIC will print the answers to arithmetical problems in response to commands such as

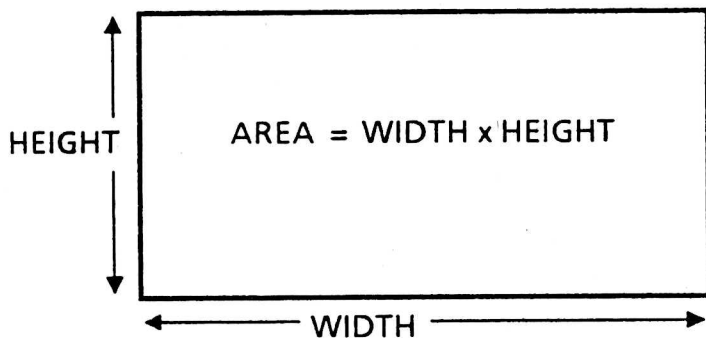
```
PRINT 3+5
```

This sort of instruction can be included in a program like this:

```
10 CLS
20 PRINT "3+5 = " 3+5
```

but it would be difficult to make much use of this for working out your personal finances.

What gives a computer the power to perform complex data processing (or 'number crunching') tasks is its ability to do algebra: to use names as symbols for numbers. This means that programs can be written with names to symbolise numbers, and then **RUN** with different numbers attached to the names. These number names are called **VARIABLES** because they represent varying numbers. Let's try an example.



The area of a rectangle is equal to the width of the rectangle multiplied by the height. By using names instead of numbers we can write a program to work out the area of any rectangle. For example:

```
10 WIDTH = 8
20 HEIGHT = 12
30 AREA = WIDTH * HEIGHT
40 PRINT "AREA = " AREA
```

This program, when **RUN**, will print the area of a rectangle 12 inches by 8 inches. By changing the

numbers in lines 10 and 20 we can obtain the area of a rectangle of any size.

VARIABLE NAMES

Any number of letters can be used in a variable name, but only the first two characters of the name are considered by the ORIC; the rest are ignored. This means for example that if you use the names

```
FR
FRUIT
FRIEDEGG
```

the computer will treat them as the same variable.

All variable names must begin with a letter, but the other characters may be numbers, for example A1, EGG7, LAST1. There must not be any spaces in a variables name: this will confuse the ORIC.

NOTE You cannot use as a variable name any of the BASIC command words, or any name which includes a command word. If you do, your program will not run, but will give **SYNTAX ERROR** messages wherever the name is used. For example, **LENGTH** contains **LEN**, and **BREADTH** contains **READ**, both of which are BASIC words. (See Appendix 1 for a full list of all the BASIC reserved words.)

TYPES OF VARIABLE

There are three different types of variable, two of which represent numbers, the third representing words. The types which represent numbers are real and integer variables; the type representing words are called string variables because they contain sequences or 'strings' of characters. The variables in the program above are real variables.

Real Variables

These are used to represent numbers and can have any value, whole numbers or fractions. Real variables should be used in all arithmetical programs.

Features

Integer Variables

These can only represent whole numbers (or integers), not fractions. They may be used when counting events or objects, or to store constant integer values. For example, a program which kept a record of the number of chocolate bars in stock at a shop could use integer variables to count them, but the takings of the shop would have to be stored in real variables, or fractions of pounds (i.e. pennies) would be ignored.

Integer variables must be identified by the % sign after the variable name, as in CHOCBARS%.

So numbers can be represented in two ways: by real variables and integer variables. Real variables can have any value. For example:

```
REAL = 3.72
SIZE = 87.3 * 2.5
```

are both real variables. Integer variables must be whole numbers:

```
NUMBER% = 3+6
COUNT% = COUNT% + 1
CHOCBARS% = 1000000
```

If you try to give an integer variable a fractional value only the whole number part is stored. So

```
10 NUMBER% = 7.5
20 PRINT NUMBER%
```

will print the value 7.

String Variables

A string variable is used to store words or letters. The variable name for a string variable must be followed by a dollar sign (\$) to distinguish it from the other types of variable. Examples are:

```
NAME$ = "WINSTON CHURCHILL"
STRING$ = "ABCDEFGH"
```

Note that the letters defining the variable contents are enclosed by quotation marks.

The maximum number of characters a string variable can hold is 255. If you try to make a string longer than that the ORIC will display the message 'STRING TOO LONG ERROR'.

NOTE It is possible to have variables of different types with the same name, such as:

```
NAME = 87.76
NAME% = 3
NAME$ = "GEORGE WASHINGTON"
```

The computer will not be confused, but you might, so be careful.

ARRAYS

Each of the three types of BASIC variable, real, integer and string, may be used in an array form. In normal use a variable name represents one stored number or string. In an array, the variable name represents a collection of stored information, with one or more reference numbers identifying the individual items. An array can be pictured as a collection of boxes. Here is a diagram of an array which uses one reference number.

The array is a string array and has 6 elements numbered from 0 to 5. The array would be used like this:

REFERENCE NUMBER	CONTENTS
0	LUCY
1	JULIE
2	MIKE
3	DAVE
4	CAROL
5	SUE

Array NAME\$

```
10 NAME$(0) = "LUCY"
20 NAME$(1) = "JULIE"
30 NAME$(2) = "MIKE"
40 NAME$(3) = "DAVE"
50 NAME$(4) = "CAROL"
60 NAME$(5) = "SUE"

100 FOR N = 0 TO 5
110 PRINT N, NAME$(N)
120 NEXT N
```

As mentioned above, an array may have more than one reference number. The number of reference numbers per item is called the number

Features

of **dimensions** of the array. Here is a program using an array with two reference numbers per item:

```
10 DIM P$(2,2)
20 P$(0,0) = "FLORENCE"
30 P$(0,1) = "NIGHTINGALE"
40 P$(1,0) = "GEORGE"
50 P$(1,1) = "WASHINGTON"
60 P$(2,0) = "ISAAC"
70 P$(2,1) = "NEWTON"

100 FOR A = 0 TO 2
110 FOR B = 0 TO 1
120 PRINT P$(A,B); " ";
130 NEXT B
140 PRINT
150 NEXT A
```

Here again the array can be thought of as a collection of storage boxes, this time laid out in a grid as shown below.

	B = 0	B = 1
A = 0	FLORENCE	NIGHTINGALE
A = 1	GEORGE	WASHINGTON
A = 2	ISAAC	NEWTON

Array P\$

You can use arrays with more than two dimensions, but you will find the ORIC's memory will not hold arrays of more than about four dimensions (the exact number depends on the number of elements in each dimension). The maximum allowed value of each reference number may be anything up to several thousand, but this depends on how much memory is occupied by the program itself, and by the other variables of the program.

A **DIM** command must be included at the beginning of any program which will use arrays of more than one dimension, or with more than eleven elements (0-10). The command tells the ORIC to reserve memory space for the array. You do not need a **DIM** command if you are using one-dimensional arrays with no more than eleven elements numbered from 0 to 10, as the ORIC is able to handle these automatically.

GETTING VALUES INTO PROGRAMS

It would be very inconvenient to have to alter a program to make it handle different numbers, so there are a number of instructions which allow numbers or letters to be given to a program while it is running. The first of these is **INPUT**.

When the computer finds an **INPUT** statement in a program, a question mark is displayed on the screen. The computer waits for you to type in a number or letter string, which it will then store in a variable before continuing with the rest of the program. Type **NEW** (RETURN) and then try this example:

```
10 INPUT NUMBER
20 PRINT NUMBER
```

When you **RUN** this program you will see a question mark on the screen. Type a number and press RETURN: the number is printed.

NOTE If you typed anything other than a number the computer would print the message 'REDO FROM START'. It would then display another question mark and wait for you to type the number in again. This is because the variable **NUMBER** is a real variable and can only store numbers. If you press RETURN without typing anything, another question mark is displayed on the next line and the ORIC continues to wait for you to enter a number.

The **INPUT** instruction can handle strings too, if you specify a string variable in the **INPUT** command:

```
10 INPUT NAMES$
20 PRINT NAMES$
```

This program will accept both numbers and letters and store them in a string variable.

Messages can be printed before the question mark to tell the person using the program what to type in. Change line 10 to:

```
10 INPUT "TYPE A WORD"; NAMES$
```

and **RUN** the program again. You can use any message you like, but it must be contained within

Features

quotation marks, and there must always be a semicolon between the message and the variable which will store the number.

It is possible to use one **INPUT** command to input two or more numbers or strings.

```
10 INPUT "NAME, AGE"; NAME$, AGE
20 PRINT NAME$
30 PRINT AGE
```

The variables must have commas separating them in the **INPUT** command; and when you type in the information you must use a comma to separate the items. If you press RETURN before entering all the items the ORIC will print two question marks and wait for the remaining items. If you enter too many items the surplus ones will be ignored and the ORIC will print '?EXTRA IGNORED'.

Let's use what we know about variables and **INPUT** to improve the area program. Clear the ORIC memory by typing **NEW** (RETURN), and then type in this program.

```
10 REM IMPROVED AREA PROGRAM
20 CLS
30 INPUT "ENTER WIDTH"; WIDTH
40 INPUT "ENTER HEIGHT"; HEIGHT
50 AREA = WIDTH * HEIGHT 'FIND AREA
60 PRINT: PRINT "AREA = " AREA
```

This program will read the two numbers you type in for width and height and print the area of the rectangle. Using **INPUT** has made the program much more flexible: we don't have to alter the program to use different numbers.

REMARKS

Line 10 of the last program is a **remark** or **comment** statement. These are used to hold comments, notes and titles to make the purpose of a program and the way in which it works clear to someone reading the listing. Remarks are identified by the word **REM** before the remark and have no effect when the program is **RUN**.

Comments may be added to the ends of program lines, as in line 50. The apostrophe (') indicates the end of the instruction and the beginning of the

comment: everything after the apostrophe is ignored by the ORIC when running the program. You should always put plenty of remarks in programs, because although you may understand how a program works when you write it, three months later you will have forgotten. If you need to modify a program at some time after writing it, remarks will make it much easier to remember how the program works.

SUMMARY

PROGRAMS

A **program** is a sequence of computer instructions, identified by numbers.

LIST displays the lines of a program.

RUN starts the execution of a program.

NEW deletes a program from the computer's memory.

VARIABLES

Variables represent numbers and words in programs.

There are three types of variable:

Real representing any numbers.

Integer representing whole numbers, and distinguished by % at the end of the variable name.

String representing words, and distinguished by \$ after the name.

INPUT is used to enter number or words into variables while a program is running.

Variables may be collected in arrays, with the same variable name, but distinguishing reference numbers.

REMARKS

REM is used to put remarks into programs for the programmer's benefit.

Comments may be added to the ends of program lines following an apostrophe (').

NEXT ISSUE: We will look at Low and Hi-Res Graphics.

Take Home Pay Calculation

By D. Lam

This program applies to the majority of people in employment irrespective of sex and the level of income. Having answered a few straightforward questions, the taxpayer should be able to check or confirm his/her take home pay.

Glossary:

- (1) GP=Gross Annual Salary
- (2) PA=Personal Allowance
- (3) TI=Taxable Income
- (4) TP=Total Pension Contribution
- (5) TN=Total National Insurance
- (6) TL=Total Tax Liability
- (7) TH=Take Home Pay for 83/84
- (8) MH=Monthly Take Home Pay

Note: For elderly taxpayers (aged sixty-five or over), replace the figure 1780 in lines 110 & 140 by 2360 AND the figure 2790 in line 150 by 3750.

Example of Screen Display

NAME: A. N. OTHER
TAX YEAR 83/84

YOUR GROSS SALARY	£20000
LESS:	
PENSION CONTRIBUTION	£600
NATIONAL INSURANCE	£873
INCOME TAX LIABILITY	£5184

YOUR NET TOTAL PAY FOR YEAR	£13343

YOUR MONTHLY TAKE HOME PAY=	£1112



PLEASE CHECK WITH YOUR PAYROLL OFFICER IF YOUR ACTUAL PAY
DIFFERS FROM ABOVE CALCULATION. THANK YOU.

Calculation Listings

```

10 REM TAKE HOME PAY CALC
15 REM FOR TAX YEAR 83/84
20 REM BY DAVID LAM
30 CLS: PRINT
40 INPUT "YOUR NAME, PLEASE"; A$
45 PRINT
50 PRINT "DO YOU KNOW YOUR
   P.A.Y.E. CODE NUMBER (Y OR N)";
52 INPUT B$: PRINT
54 IF LEFT$(B$,1) <> "N" AND
   LEFT$(B$,1) <> "Y" THEN PRINT "Y
   OR N"; : GOTO 52
60 IF LEFT$(B$,1) = "N" THEN 100
70 IF LEFT$(B$,1) = "Y" THEN PRINT
   "ENTER YOUR CODE NUMBER";
75 INPUT C$
80 PA=VAL(C$)*10
90 PRINT:GOTO 170
100 PRINT "YOUR SEX (M OR F)";
105 INPUT D$:PRINT
108 IF LEFT$(D$,1)<>"F" AND
   LEFT$(D$,1)<>"M" THEN PRINT "M
   OR F"; : GOTO 105
110 IF LEFT$(D$,1)="F" THEN
   PA=1780:GOTO 170
120 IF LEFT$(D$,1)="M" THEN PRINT
   "ARE YOU SINGLE/MARRIED (S OR
   M)";
130 INPUT E$: PRINT
135 IF LEFT$(E$,1)<>"S" AND
   LEFT$(E$,1)<>"M" THEN PRINT "S
   OR M"; : GOTO 130
140 IF LEFT$(E$,1)="S" THEN PA=1780
150 IF LEFT$(E$,1)="M" THEN PA=2790
170 INPUT "ENTER GROSS ANNUAL
   SALARY £"; GP
180 PRINT
190 PRINT "ON COMPANY PENSION
   PLAN (Y OR N)";
195 INPUT F$: PRINT
198 IF LEFT$(F$,1) <> "N" AND
   LEFT$(F$,1) <> "Y" THEN PRINT "Y
   OR N"; : GOTO 195
200 IF LEFT$(F$,1)="N" THEN TP=0:
   GOTO 300
210 IF LEFT$(F$,1)="Y" THEN PRINT
   "ENTER MONTHLY DEDUCTION £";
220 INPUT P
230 TP=INT(P*12+0.5)
240 CA=140.83*0.09
250 CB=(GP/12-140.83)*0.0685

```

```

260 IF>CB 60.11 THEN CB=-60.11
270 CO=CA+CB
280 TN=INT(CO*12+0.5)
290 GOTO 330
300 CI=GP/12*0.09
310 IF CI>91.66 THEN CI=91.66
320 TN=INT(CI*12+0.5)
330 TI=GP-(PA+TP)
340 IF TI<14600 THEN TL=TI*0.30
350 IF TI>14600 AND TI<=17200 THEN
   TL=4380+(TI-14600)*0.40
360 IF TI>17200 AND TI<=21800 THEN
   TL=5420+(TI-17200)*0.45
370 IF TI>21800 AND TI<=28900 THEN
   TL=7490+(TI-21800)*0.50
380 IF TI>28900 AND TI<=36000 THEN
   TL=11040+(TI-28900)*0.55
390 IF TI>36000 THEN TL=14945+(TI-
   36000)*0.60
400 TH=GP-(TP+TN+TL)
410 MH=INT(TH/12+0.5)
420 WAIT 20:CLS:PRINT
430 PRINT "NAME:";A$:PRINT
440 PRINT "TAX YEAR 83/84":PRINT
450 PRINT "YOUR GROSS SALARY
   ----- £";GP
460 PRINT
465 PRINT "LESS:"
470 PRINT "PENSION CONTRIBUTION
   ----- £";TP
480 PRINT "NATIONAL INSURANCE
   ----- £";TN
490 PRINT "INCOME TAX LIABILITY
   ----- £";INT(TL+0.5)
500 PRINT
510 PLOT 29,11, "-----"
520 PRINT "YOUR NET TOTAL PAY FOR
   YEAR £";TH
530 PLOT 29,13, "===== "
540 PRINT : PRINT
550 PRINT "YOUR MONTHLY TAKE
   HOME PAY = £";MH
560 PRINT : PRINT
570 PRINT "PLEASE CHECK WITH YOUR
   PAYROLL OFFICER IF YOUR
   ACTUAL PAY";
580 PRINT "DIFFERS FROM ABOVE
   CALCULATION. THANK YOU"
600 END.

```

And Forthly . . . Money Manager

By W. Cain

Objective

To provide an easily maintained database of transaction within a current account.

To enable a future cash flow forecast to be generated between start and end dates given on initial account balance. This combination of facilities provides, in addition to the cash flow management ability, the possibility of running 'What-if' budgets i.e. what-if the wife stopped working?!



User Interface

1. The Database

The database allows a maximum of 48 records and is backed-up by cassette as 3 Forth 'screens'.

Each record contains the following fields:

1. Transaction title – any

- meaningful title of up to 16 characters.
2. Amount – the value of the transaction (always entered as a positive number). Allowed formats are £xxxx or £xxxx.x or £xxxx.xx.
 3. Credit/Debit – defines the direction of the transaction. Enter C for credit D for debit.
 4. Start Date – the start date of this type of transaction. Entered as DD/MM/YY.
 5. End Date – the end date of this type of transaction. Entered as DD/MM/YY.
 6. Period – the interval between transactions of this type. Entered as DD/MM/YY i.e a weekly transaction is entered as 7/0/0.
- Records as manipulated by the following commands:
- A. ENTER – allows prompted entry of a complete record.

B. REMOVE – asks for the record key (transaction title), displays the record and asks for a confirming Y or N to complete the removal.

C. CHANGE – asks for the record key (transaction title), displays the record and asks for selection of the field to be changed by number. Entry of changes is identical to entry of fields in the ENTER command. Multiple changes must be done by multiple CHANGE commands.

D. DISPLAY – displays the records in the database in record order. Suspends at the end of each screen until de-suspended by any key press.

E. DUMP – asks that you position cassette tape and set REC/PLAY then dumps database to cassette as 3 FORTH screens.

When the first attempted access to the database is made during a run of the program it will suspend and ask whether you want to use an existing database or set up a new one. A previously 'DUMPed' database may be loaded at this point by positioning a cassette and de-suspending by any key press.

(HINT – if you want to forget about the current database *during* a session you can enter 0 LOADED!)

2. Data Analysis

The cash-flow predictor facility is initiated by the command FORECAST.

This will ask for the date of the start of the forecast period (enter as DD/MM/YY), the date of the end of the forecast period (enter as DD/MM/YY) and the assumed initial account balance. This may be entered as a negative number signifying a current overdraft.

After a short period during which transaction dates are advanced into the forecast window the transactions will be shown in date order in a format which approximates to a bank statement. The listing will suspend at the end of a screen and is de-suspended by any key press.

3. Reports

If a printer is supported the following commands will generate hard-copy.

LIST-ITEMS – lists records in the database (similar to DISPLAY).

FORECAST-REPORT – lists transactions in time order through the specified forecast period (similar to FORECAST).

Hints for Efficient Running

1. Remove 'dead' records i.e. those whose finish date is prior to any forecast window.
2. Make the start dates as close to any anticipated forecast window as you can. This will save time when these transactions are advanced into the forecast window.
3. 'Difficult' transactions such as electricity bills whose size depends on the season may be entered as separate 'annual' transactions e.g. ELEC WINTER, ELEC SPRING etc.
4. *Don't* put leading spaces in the transaction title. This will cause it to be (at least) ignored or (at worst) overwritten.

Apologies

Dates in the next century are output as DD/MM/YY e.g. 2nd December 2014 is 2/12/114.

Leap years are not handled.

Setting-up

The program is supplied as seven fairly dense screens of Forth source.

They can be loaded on top of any forth system which contains at least

1. Basic ORIC Forth
2. The word – TEXT from ORIC Forth Editor
3. Screens 1 and 4 of the ORIC Forth extensions.

Compiling time is about 40 seconds.

Database commands are practically instantaneous.

In forecast mode transactions are processed and sorted at an average rate of two every 3 seconds.

If a printer is available a further two screens may be loaded which will support the LIST-ITEMS and FORECAST-REPORT commands. Printer controls are set for the STAR DP510.

Implementation

1. Data

New data type n VAR creates variable allocation for 2n bytes and returns variable address.

Used as 3 VAR for dates DAYS/MONTHS/YEARS

Used as 2 VAR for current balance – double length signed number.

New data type 12CLIST (circular list of 12 constant values) used as n CLIST returning the value of the list indexed by n modulo 12. Used in calendar calculations – days in month n.

Record Structure

	Displacement from Origin	Length (bytes)	Type
TITLE	0	16	byte string
AMOUNT	16	4	unsigned integer
C/D	20	1	byte
START	21	6	3 VAR
FINISH	27	6	3 VAR
STEP	33	6	3 VAR
NEXT	39	6	3 VAR
SPARE	45	18	unassigned

Records are mapped as 2 per 128 byte block.

Mini-disc allocation starts at block 8 and runs to block 32.

2. Program

– RECORD	addr	Loads block containing record number #REC and returns start address of record.
– FREE	–	Sets #REC to the index of the first free record (first byte is 'space').
– WIPE	–	Sets blocks 8–32 to spaces.
addr DATE	–	Prints 3VAR at addr as DD/MM/YY.
n PAGE?	–	Suspends on key press when #N = n with message PRESS ANY KEY TO CONTINUE.
addr SETNXT	–	Moves START field into NEXT for record at addr.

Software

- addr UPDNXT - Adds STEP to NEXT for record at addr.
addr1 addr2 LATERt/f For records at addr1 and addr2 returns true if NEXT of record 2 is later than NEXT of record 1.
- SET1ST - For all valid records sets NEXT to the first date equal to or later than the date in 3 VAR NOW.
- EARLIEST addr Returns addr of record with the earliest NEXT date.

Other words are more application dependent and deal with I/O and screen display.

And Forthly . . . Listing

```
SCREEN1
0 ( DATABASE PRIMITIVES)
1 0 CONSTANT TITLE 16 CONSTANT AMOUNT 20 CONSTANT C/D
2 21 CONSTANT START 27 CONSTANT FINISH 33 CONSTANT STEP
3 39 CONSTANT NEXT 45 CONSTANT SPARE 64 CONSTANT R-LEN
4 B/BUF R-LEN / CONSTANT REC/BLK 8 CONSTANT FILES 0 VARIABLE EREC
5 32 CONSTANT MBLK : TOP 0 EREC ! ; : DOWN 1 EREC +! ;
6 MBLK FILES - REC/BLK * CONSTANT MAXRECS
7 0 VARIABLE LOADED 0 VARIABLE EN 0 VARIABLE EREC
8 : D@ DUP @ SWAP 2+ @ ; : .STERL DABS 100 M/ 4 .R ." ." 2 .R ;
9 : D* DUP ROT U* DROP ROT ROT U* ROT 0 SWAP D+ ;
10 : GET TIB @ OVER BLANKS QUERY TIB @ ROT ROT CMOVE 0 TIB @ C! ;
11 : WIPE MBLK FILES DO 1 BLOCK B/BUF BLANKS UPDATE LOOP ;
12 : INIT EMPTY-BUFFERS CR
13 ." PRESS N FOR NEW FILE,SPACE FOR EXISTING DATA" KEY 78 =
14 IF WIPE ELSE CR ." LOAD DATA TAPE,ANY KEY TO CONTINUE" KEY
15 DROP 1 3 CLOAD ENDIF 1 LOADED ! ; -->
```

```
SCREEN2
0 ( BUDGET DATABASE)
1 : RECORD LOADED @ 0= IF INIT ENDIF EREC @ REC/BLK /MOD
2 FILES + BLOCK SWAP R-LEN * + ;
3 : FREE 1 MAXRECS 0 DO 1 EREC ! RECORD C@ 32 = IF DROP 0
4 LEAVE ENDIF LOOP IF ." FILE FULL" ABORT ENDIF ;
5 : DUMP ." SET DATA TAPE REC/PLAY AND PRESS ANY KEY " KEY
6 DROP FLUSH 1 3 CSAVE CR ;
7 : TITLE? TITLE + CLS CR ." TRANSACTION NAME?" 16 GET ;
8 : AMOUNT? AMOUNT + CR CR ." HOW MUCH(X.X) ?..E" QUERY 0 WORD
9 HERE NUMBER DPL @ 2 < IF 10 D* ENDIF DPL @ 1 < IF 10 D* ENDIF
10 ROT DUP ROT SWAP 2+ ! ! ( STORE IN AMT LO THEN HI) ;
11 : C/D? C/D + CR CR ." CREDIT(C) OR DEBIT(D)?" 1 GET ;
12 : IND QUERY 3 0 DO 47 WORD HERE NUMBER DROP OVER ! 2+ LOOP ;
13 : START? CR CR ." START DATE(DD/MM/YY)? " START + IND DROP ;
14 : FINISH? CR CR ." FINISH DATE(DD/MM/YY)? " FINISH + IND DROP ;
15 -->
```

```
SCREEN3
0 ( DATABASE AND LISTING COMMANDS)
1 : STEP? CR CR ." TIME BETWEEN PAYMENTS" CR ." (DAYS/MONTHS/"
2 ." YEARS)" STEP + IND DROP ;
3 : ENTER FREE RECORD DUP TITLE? DUP AMOUNT? DUP C/D? DUP START?
4 DUP FINISH? STEP? UPDATE ;
5 : HEADING CR ." START" 4 SPACES ." FINISH" 3 SPACES
6 ." PERIOD" CR ;
7 : .DATE 3 0 DO DUP @ 2 .R 2+ 1 2 < IF ." /" THEN LOOP DROP
8 SPACE ;
9 : .PERIOD STEP + DUP @ DUP IF ." DAYS" ELSE DROP ENDIF 2+ DUP
10 @ DUP IF ." MONTHS" ELSE DROP ENDIF 2+ @ DUP IF ."
11 ." YEARS" ELSE DROP ENDIF ;
12 : .DATES DUP START + .DATE DUP FINISH + .DATE .PERIOD ;
13 : PPREC CR DUP 16 TYPE 27 EMIT DUP C/D + C@ 3 - EMIT
14 ." _" DUP AMOUNT + D@ .STERL HEADING .DATES CR ;
15 -->
```

ORIC-1 FIG-FORTH

And Forthly . . . Listing

```
SCREEN4
0 ( LIST REMOVE AMEND FACILITIES)
1 : PAGE? EN @ SWAP = IF 0 EN ! CR ." PRESS ANY KEY TO CONTINUE"
2 KEY DROP GTC DROP CLINE 11 EMIT ENDIF ;
3 : DISPLAY CLS TOP 0 EN ! MAXRECS 0 DO RECORD DUP C@ 32 = 0= IF
4 PPREC 1 EN +! 6 PAGE? ELSE DROP THEN DOWN LOOP CR ." END LIST" ;
5 : NFIND 0 QUERY 0 WORD MAXRECS 0 DO I EREC ! HERE COUNT
6 RECORD EDITOR -TEXT FORTH DUP IF LEAVE ELSE DROP ENDIF
7 LOOP IF ELSE 1 CR ." NOT IN FILE" ENDIF ;
8 : GETREC CLS CR ." TRANSACTION TITLE?" NFIND IF 0 ELSE RECORD
9 DUP PPREC CR ." IS THIS THE ONE YOU MEAN?(Y/N)" KEY 89 = IF
10 1 ELSE DROP 0 CR ." STET" ENDIF ENDIF ;
11 : REMOVE GETREC IF R-LEN BLANKS UPDATE CR ." REMOVED" THEN ;
12 : CHANGE GETREC IF CR ." WHICH FIELD DO YOU WANT TO CHANGE?" CR
13 ." TYPE" CR ." 1(TITLE)" CR ." 2(AMOUNT)" CR ." 3(C/D)" CR
14 ." 4(START)" CR ." 5(FINISH)" CR ." 6(PERIOD)" CR INE CASE
15 1 OF TITLE? ENDOF 2 OF AMOUNT? ENDOF 3 OF C/D? ENDOF -->
```

```
SCREEN5
0 ( BUDGET DBASE ANALYSIS)
1 4 OF START? ENDOF 5 OF FINISH? ENDOF 6 OF STEP? ENDOF
2 ENDCASE UPDATE CR ." CHANGE DONE" ENDIF ;
3 : VAR <BUILDS 2 * ALLOT DOES> ; 2 VAR BAL 3 VAR NOW
4 3 VAR FIN 3 VAR REF
5 : 12CLIST <BUILDS 31 , 31 , 28 , 31 , 30 , 31 , 30 , 31 , 31
6 , 30 , 31 , 30 , DOES> SWAP 12 MOD 2 * + @ ; 12CLIST DAYSIN
7 : SETNXT DUP START + SWAP NEXT + 6 CMOVE UPDATE ;
8 : UPDNXT DUP NEXT + SWAP STEP + OVER @ OVER @ + 3 PICK 2+ @
9 BEGIN OVER OVER DAYSIN > WHILE DUP DAYSIN ROT SWAP - SWAP 1+
10 REPEAT SWAP 4 PICK ! ROT 2+ ROT 2+ ROT OVER @
11 + 1 - 12 /MOD SWAP 1+ 4 PICK ! ROT 2+ ROT 2+ ROT SWAP @
12 + SWAP +! UPDATE ;
13 : LATER OVER 4 + @ OVER 4 + @ - -DUP 0= IF OVER 2+ @ OVER 2+
14 @ - -DUP 0= IF OVER @ OVER @ - 0< IF 1 ELSE 0 THEN ELSE
15 0< IF 1 ELSE 0 THEN THEN ELSE 0< IF 1 ELSE 0 THEN THEN -->
```

```
SCREEN6
0 ( CONTINUE BUDGET DATABASE ANALYSIS) ROT DROP SWAP DROP ;
1 : EARLIEST 999 REF 4 + !
2 TOP BEGIN RECORD DUP C@ 32 > IF DUP NEXT + REF
3 LATER IF EREC @ EREC ! DUP NEXT + REF 6 CMOVE ENDIF ENDIF DROP
4 1 EREC +! EREC @ MAXRECS = UNTIL EREC @ EREC ! RECORD ;
5 : SET1ST MAXRECS 0 DO I EREC ! RECORD DUP C@ 32 > IF DUP
6 SETNXT BEGIN DUP NEXT + NOW LATER WHILE DUP UPDNXT REPEAT
7 ENDIF DROP LOOP ;
8 : FROMTO? CR ." FROM WHAT DATE(DD/MM/YY)?" NOW IND DROP CR CR
9 ." TO WHAT DATE(DD/MM/YY)?" FIN IND DROP CR CR ." WITH A "
10 ." STARTING BALANCE OF..." BAL AMOUNT - AMOUNT? ;
11 : .XSAC CR DUP NEXT + .DATE CR DUP 16 TYPE 27 EMIT DUP
12 C/D + C@ 3 - EMIT ." _" DUP AMOUNT + D@ OVER OVER .STERL
13 ROT C/D + C@ 68 = IF DMINUS THEN BAL D@ D+ OVER OVER BAL 2+ !
14 BAL ! DUP 27 EMIT 0< IF 65 EMIT DABS ELSE 64 EMIT THEN
15 ." £" .STERL ; -->
```

ORIC-1 FIG-FORTH

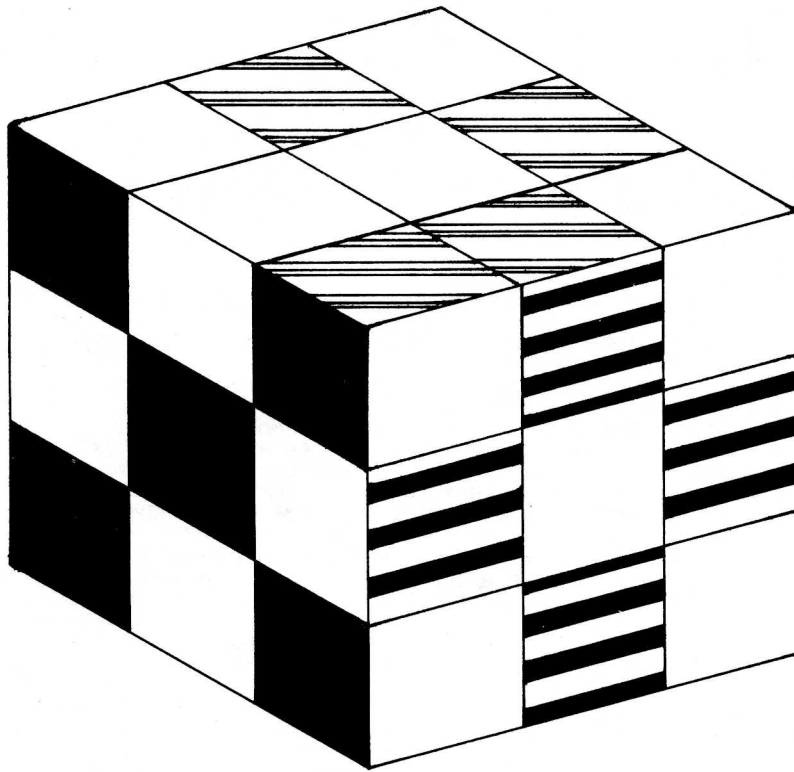
```
SCREEN7
0 ( TOP LEVEL FORECAST WORD)
1 : FORECAST CLS FROMTO? SET1ST 0 EN ! CLS BEGIN EARLIEST DUP
2 NEXT + FIN LATER WHILE DUP NEXT + SWAP FINISH + LATER IF
3 DUP .XSAC UPDNXT 1 EN +! 12 PAGE? ELSE UPDNXT THEN REPEAT DROP ;
4 : PNAME HERE 14 BLANKS 0 WORD HERE 1+ 5162 13 CMOVE ;
5 PNAME MONEY MANAGER
6 FORTH DEFINITIONS DECIMAL
7 LATEST 12 +ORIGIN !
8 HERE 28 +ORIGIN !
9 HERE 30 +ORIGIN !
10 HERE FENCE !
11 ." MONEY MANAGER LOADED"
12 ;S
13
14
15
```

ORIC-1 FIG-FORTH

Software

Rubik Cube

By J. Airey



This program for the 48K ORIC simulates a rubik cube and shows what can be achieved with serial attributes. It shows the 4 sides around the side, the top side and the bottom side soaking through the cube.

Type in the program but save it before you run it, twice to make sure.

Run the program and you should see the instruction page, press the space bar and you will see the next page of instructions.

Press any key and the cube will then be drawn in basic. To test the program when it's drawn, press I and you should see the instructions again. If pressing I doesn't work and the ORIC has crashed then you have mistyped a data line.

Go through the instructions again and this time the cube will be drawn in machine code in about a half of a second.

Now, a small explanation of the machine code.

Locations #229 and #22A address of USER IRQ routine. Use these to point to your interrupt routine. End your routine with JMP #ECO3 or #EDO9.

Locations #2F5 and #2F6, address of ! command. Saves having to use call whatever and makes it easier to relocate machine code.

To convert program to 16K, remove the screen m.c. Data alter the checksum accordingly and substitute GOSUB2000 with redrawing the screen in basic. Also take out all the !'s.

Software

Rubik Cube Listing

```
10 HIMEM#7400:DOKE#229,REC03
20 REM***INSTRUCTIONS***
30 TEXT:INK7:PAPER0:POKE#18,CLS
40 GOSUB900
50 PLOT10,4,"By John Airey"
60 PLOT3,6,"In this program you move the cube"
70 PLOT1,7,"by using a PASS joystick or keys."
80 PLOT1,9,"There are 18 different ways to turn"
90 PLOT1,10,"a cube i.e. 9 forward 9 backward"
100 PLOT1,11,"so I divided it into nine different"
110 PLOT1,12,"rotations and I used a + or a - to"
120 PLOT1,13,"show which way you are turning the"
130 PLOT1,14,"cube"
140 PLOT1,16,"position arrow (") under the number"
150 PLOT1,17,"of the direction you require"
160 AS="move arrow left":X=20:Y=19
170 PLOT1,Y,"left arrow":PLOT1,20,"joystick left"
180 GOSUB970
190 AS="move arrow right":Y=21
200 PLOT1,Y,"right arrow":PLOT1,22,"joystick right"
210 GOSUB970
220 AS="execute move":Y=23
230 PLOT1,Y,"space bar: PLOT1,24,"fire button"
240 GOSUB970
250 AS="PRESS SPACE BAR TO CONTINUE":Y=25:X=5
260 GOSUB970
270 GEPAS:IFAS<>" THEN270ELSECLS
280 GOSUB900:AS="change to plus":Y=5:X=20
290 PLOT1,Y,"+ key":PLOT1,6,"joystick forward"
300 GOSUB970
310 AS="change to minus":Y=9
320 PLOT1,Y,"- key":PLOT1,10,"joystick back"
330 GOSUB970
340 AS="scramble cube":Y=13
350 GOSUB970
360 AS="S key":X=1
370 GOSUB970
380 AS="I key":Y=17
390 GOSUB970
400 AS="return to menu":X=20
410 GOSUB970
420 AS="PRESS ANY KEY TO START CUBING":Y=25:X=4
430 GOSUB970
440 GEPAS:HIRES
445 REM***IF SECOND PASS THEN***
446 REM***GOTO MAIN LOOP***
450 IFPEEK(4200)=48:THENGOSUB2000:GOTO1020
455 REM***ELSE RESET CUBE***
460 GOTO1000
470 AS="R U B I K C U B E"
910 PLOT10,1,AS:PLOT10,2,AS
920 PLOT8,1,4:PLOT8,2,4
930 PLOT1,1,18:PLOT9,2,18
940 FORA=0TO20
950 B=RND(1)*6+1:PLOT8,A,B
960 NEXT:RETURN
970 PLOT,Y,AS:PLOTX,Y+1,AS
980 PLOTX-1,Y+1,PLOTX-1,Y+1,10
990 RETURN
995 REM***MAKE SPACE FOR VARIABLES***
1000 DIM(54),Y(54),CO(54)
1010 GOSUB3000:GOSUB2000
1020 AS="KEYS:IFAS=" ANDDEK(400)=0:THEN1020
1025 REM***CHANGE TO PLUS***
1030 IFAS="ORPEEK(400)=0:THEN26=1:GOTO2040
1035 REM***MOVE ARROW LEFT***
1040 IFAS=CHR$(8) ANDA>1:THENGOSUB2020:A=A-1:GOSUB2020
1050 IFPEEK(400)=1:LANDA>1:THENGOSUB2020:A=A-1:GOSUB2020
1055 REM***MOVE ARROW RIGHT***
1060 IFAS=CHR$(8) ANDA<9:THENGOSUB2020:A=A+1:GOSUB2020
1070 IFPEEK(400)=2:LANDA<9:THENGOSUB2020:A=A+1:GOSUB2020
1075 REM***MOVE CUBE***
1080 IFAS="ORPEEK(400)=16:THEN110
1085 REM***SCRAMBLE CUBE***
1090 IFAS="S":THENGOSUB4000
1095 REM***BACK TO INSTRUCTIONS***
1100 IFAS<>" THEN1020ELSE30
1110 ONAGOSUB5000,5100,5200,5300,5400,5500,5600,5700,5800
1120 GOSUB2090:GOSUB2160:GOTO1020
1130 AS="KEYS:IFAS=" ANDDEK(400)=0:THEN1130
1135 REM***CHANGE TO MINUS***
1140 IFAS="ORPEEK(400)=4:THEN26=0:GOTO2040
1145 REM***MOVE ARROW LEFT***
1150 IFAS=CHR$(8) ANDA>1:THENGOSUB2020:A=A-1:GOSUB2020
1160 IFPEEK(400)=1:LANDA>1:THENGOSUB2020:A=A-1:GOSUB2020
1165 REM***MOVE ARROW RIGHT***
1170 IFAS=CHR$(8) ANDA<9:THENGOSUB2020:A=A+1:GOSUB2020
1180 IFPEEK(400)=2:LANDA<9:THENGOSUB2020:A=A+1:GOSUB2020
1185 REM***MOVE CUBE***
1190 IFAS="ORPEEK(400)=16:THEN120
1195 REM***SCRAMBLE CUBE***
1200 IFAS="S":THENGOSUB4000
1205 REM***BACK TO INSTRUCTIONS***
1210 IFAS<>" THEN1020ELSE30
1220 FORC=1TO3
1230 ONAGOSUB5000,5100,5200,5300,5400,5500,5600,5700,5800
1240 NEXT:GOSUB2090:GOSUB2160:GOTO1130
2000 !:GOSUB2090
2010 A=5:24:67:25:16:RETURN
2015 REM***DRAW ARROW(")***
2020 CURSET55=(A*12)+8,3
2030 CHAR94,0,2:RETURN
2035 REM***DRAW -> SIGNS***
2040 CURSET0,0,3
2050 CHAR43,0,2
2060 CURMOV0,0,3
2070 CHAR45,0,2
2080 IFZ6=0:THEN1020ELSE1130
2085 REM***COLOUR IN ATTRIBUTES***
2090 FORB=1TO36
2100 CURSETX(B),Y(N),3
2110 FILL36,1,CO(B)
2120 NEXT:FORB=37TO54
2130 CURSETX(B),Y(D),3
2140 FILL12,1,CO(B)
2150 NEXT:RETURN
2155 REM***PRINT OUT MOVE***
2160 CURSET24,25,3:IFZ6=1:THENCHAR43,0,2:GOTO2100
2170 CHAR45,0,2
2180 CURMOV0,0,3:CHARA=0,0,2:24=24+12
2190 IFZ4=155:THENZ4=67:25=25+8
2200 IFZ5<48:THEMRETURN
2210 GOSUB2000:GOSUB2090
2220 IFZ6=1:THENGOSUB2040
2230 RETURN
2995 REM***DRAW SCREEN VIA M.C.***
3000 DEFN(X)=PEEK(537)
3010 DEFN(Y)=PEEK(538)
3020 FORA=1270TO48STEP18
3030 FORB=ATOA+11
3040 C=B+116:D=125-B
3050 E=B+108:F=233-B
3060 FORC=0TO2
3070 H=C+18
3080 CURSETB,C,3
3090 DRAWFNX(B),FNY(C+24),2
3100 CURSETD,C,3
3110 DRAWFNX(D),FNY(C+24),2
3120 CURSETF,C,3
3130 DRAWFNX(F),FNY(C+24),2
3140 CURSETG,C,3
3150 DRAWFNX(G),FNY(C+24),2
3160 CURSETH,C,3
3170 DRAWFNX(H),FNY(H+12),2
3180 CURSETI,C,3
3190 DRAWFNX(I),FNY(H+12),2
3200 C=C+36:NEXTG,A
3210 CURSET61,0,3:FILL54,1,7:CURMOV0,0,3
```

```
3220 FORA=49TO57:CHARA,0,2
3230 CURMOV12,0,3:NEXT
3240 CURSET12,191,3
3250 FORA=49TO54:CHARA,0,2
3260 CURMOV18,0,3:IFA=51:THENCURMOV0,0,3
3270 NEXT:CURMOV-6,0,3:FORA=51TO49STEP-1
3280 CHARA,0,2:CURMOV18,0,3:NEXT
3290 CURMOV0,0,3:FORA=54TO52STEP-1
3300 CHARA,0,2:CURMOV18,0,3:NEXT
3310 CURSET0,60,3:FORA=55TO57
3320 CHARA,0,2:CURMOV0,36,3
3330 NEXT:CURSET232,60,3:FORA=55TO57
3340 CHARA,0,2:CURMOV0,36,3
3350 NEXT:CURSET0,8,3:CHARA5,0,2
3360 CURSET115,8,3:CHARA94,0,2
3370 FORA=403TO485:READS:POKEA,B
3380 CS=CS+B:NEXTA:IFCS<0:1060:THENSTOP
3390 DOKE#229,403:DOKE#2F5,4450
3400 I:POKE#478,129:POKE#47A,131
3410 FORX=205TO257STEP-16:FORB=1TO3
3420 Z=Z+1:READY,C,CK(2)*X:Y(2):Y:CO(2)=C
3430 NEXTB,X
3440 FORX=437TO7STEP-18:FORB=WT02
3450 Z=Z+1:Y=(Y+18)+1
3460 X(Z)=X+164:Y(Z)=Y:CO(2)=6
3470 X(Z+9)=X:Y(Z+9)=Y:CO(2+9)=4
3480 NEXTD,X
3490 RETURN
3495 REM***JOYSTICK DATA***
3500 DATA72,173,15,3,72,173,3,3
3510 DATA72,169,192,141,3,3
3520 DATA169,128,141,15,3,32,51,4
3530 DATA141,1,4,169,64,141,15,3
3540 DATA32,51,4,141,1,4,169
3550 DATA141,3,3,104,141,15,3
3560 DATA104,76,3,236,169,0
3570 DATA141,2,4,173,15,3
3580 DATA73,59,10,10,46
3590 DATA2,4,10,46,2,4,10,46
3600 DATA2,4,10,10,46,2,4,10
3610 DATA46,2,4,173,2,4,46
3615 REM***SCREEN DATA***
3620 DATA73,111,2,201,27,208,1
3630 DATA96,169,10,141,106,2,169,0
3640 DATA133,129,133,131,169,152
3650 DATA133,130,169,192,133,132
3660 DATA162,33,169,0,177,131
3670 DATA145,129,136,208,249,198
3680 DATA130,198,132,202,208,240
3690 DATA96
3695 REM***DATA FOR CURE***
3700 DATA57,2,93,2,128,2
3710 DATA75,2,111,2,147,2
3720 DATA93,2,128,2,164,2
3730 DATA93,1,128,1,164,1
3740 DATA75,1,111,1,147,1
3750 DATA57,1,93,1,128,1
3760 DATA75,5,93,5,128,5
3770 DATA75,5,111,5,147,5
3780 DATA93,5,128,5,164,5
3790 DATA93,3,128,3,164,3
3800 DATA75,3,111,3,147,3
3810 DATA57,3,93,3,128,3
3895 REM***SCRAMBLE CUBE***
4000 FORI=1TO10:CURSET0,16,3:CHAR03,0,2
4010 B=RND(1)*8+1
4020 ONBOSUB5000,5100,5200,5300,5400,5500,5600,5700,5800
4030 NEXT:GOSUB2090
4040 RETURN
4995 REM***ROUTINES FOR TURNING CUBE***
5000 FORB=1TO3:(B)=CO(33)+B
5010 CO(33+B)=CO(51+B)
5020 CO(51+B)=CO(13+B)
5030 CO(13+B)=CO(36+B*3)
5040 CO(36+B*3)=2(B):NEXT
5050 Z(1)=CO(11):Z(2)=CO(2)
5060 CO(1)=CO(7):CO(2)=CO(4)
5070 CO(7)=CO(8):CO(4)=CO(8)
5080 CO(9)=CO(3):CO(8)=CO(6)
5090 CO(3)=2(1):CO(6)=2(2):RETURN
5100 FORB=1TO3:(B)=CO(38)+B
5110 CO(38+B)=CO(48+B)
5120 CO(48+B)=CO(16+B)
5130 CO(16+B)=CO(35+B*3)
5140 CO(35+B*3)=2(B):NEXT
5150 RETURN
5200 FORB=1TO3:(B)=CO(27)+B
5210 CO(27+B)=CO(45+B)
5220 CO(45+B)=CO(19+B)
5230 CO(19+B)=CO(34+B*3)
5240 CO(34+B*3)=2(B):NEXT
5250 Z(1)=CO(19):Z(2)=CO(20)
5260 CO(19)=CO(23):CO(20)=CO(24)
5270 CO(23)=CO(27):CO(24)=CO(26)
5280 CO(27)=CO(25):CO(26)=CO(22)
5290 CO(25)=2(1):CO(22)=2(2):RETURN
5300 FORB=1TO3:(B)=CO(24)+B
5310 CO(24+B)=CO(57+B*3)
5320 CO(57+B*3)=CO(4-B)
5330 CO(4-B)=CO(36+B)
5340 CO(36+B)=2(B):NEXT
5350 Z(1)=CO(28):Z(2)=CO(29)
5360 CO(28)=CO(34):CO(29)=CO(31)
5370 CO(34)=CO(36):CO(31)=CO(35)
5380 CO(36)=CO(30):CO(35)=CO(33)
5390 CO(30)=2(1):CO(33)=2(2):RETURN
5400 FORB=1TO3:(B)=CO(21)+B
5410 CO(21+B)=CO(56+B*3)
5420 CO(56+B*3)=CO(7-B)
5430 CO(7-B)=CO(39+B)
5440 CO(39+B)=2(B):NEXT
5450 RETURN
5500 FORB=1TO3:(B)=CO(18)+B
5510 CO(18+B)=CO(55+B*3)
5520 CO(55+B*3)=CO(18-B)
5530 CO(18-B)=CO(42+B)
5540 CO(42+B)=2(B):NEXT
5550 Z(1)=CO(15):Z(2)=CO(12)
5560 CO(15)=CO(17):CO(12)=CO(18)
5570 CO(17)=CO(13):CO(18)=CO(16)
5580 CO(13)=CO(11):CO(16)=CO(10)
5590 CO(11)=2(1):CO(10)=2(2):RETURN
5600 FORB=3TO9STEP3:(B)=CO(B-2)
5610 CO(B-2)=2(7+B)
5620 CO(7+B)=CO(16+B)
5630 CO(16+B)=CO(25+B)
5640 CO(25+B)=2(B):NEXT
5650 Z(1)=CO(48):Z(2)=CO(51)
5660 CO(48)=CO(54):CO(51)=CO(53)
5670 CO(54)=CO(52):CO(53)=CO(49)
5680 CO(52)=CO(46):CO(49)=CO(47)
5690 CO(46)=2(1):CO(47)=2(2):RETURN
5700 FORB=3TO9STEP3:(B)=CO(B-1)
5710 CO(B-1)=CO(8+B)
5720 CO(8+B)=CO(17+B)
5730 CO(17+B)=CO(26+B)
5740 CO(26+B)=2(B):NEXT
5750 RETURN
5800 FORB=3TO9STEP3:(B)=CO(B)
5810 CO(B)=CO(9+B)
5820 CO(9+B)=CO(18+B)
5830 CO(18+B)=CO(27+B)
5840 CO(27+B)=2(B):NEXT
5850 Z(1)=CO(37):Z(2)=CO(36)
5860 CO(37)=CO(39):CO(38)=CO(42)
5870 CO(39)=CO(45):CO(42)=CO(44)
5880 CO(45)=CO(43):CO(44)=CO(40)
5890 CO(43)=2(1):CO(40)=2(2):RETURN
```

Bar graph drawing program

By K. Golding

RUN'S in under a kilobyte

On running you enter how many items you have (1-7) and then the amount (up to 25) and the item name, then it draws the graphs.

```
3 POKE #26A,35
5 TEXT: PAPER 0:INK 7
10 FOR T=0 TO 7
20 B$(T)=CHR$(T+144): F$(T)=CHR$(T+128)
40 NEXT
100 CLS: PRINT
110 PRINT SPC(15)B$(7)SPC(12)B$(0)
120 PRINT SPC(5) B$(7) F$(0) "BAR CHARTS" B$(0)
130 PRINT SPC(15) B$(7) SPC(12) B$(0)
131 PRINT:PRINT "HOW MANY ITEMS";:INPUT U: IF U>7
    OR U<0 THEN POKE #268,5:GOTO 131
132 POKE#268,5
140 PRINT:PRINT "ENTER THE DATA:"
145 PRINT SPC(17)"": (1 TO 25), ITEM"
150 FOR M=0 TO U-1
155 IF PEEK(#268)>25 THEN POKE #268,18
160 INPUT ".....": A(M), A$(M)
170 IF A(M)>25 THEN EXPLODE: GOTO 1600
172 IF A(M)<0 THEN EXPLODE: GOTO 160
175 IF LEN(A$(M))>9 THEN SHOOT: GOTO 160
180 NEXT
190 POKE #268,4: FOR Q=1 TO 22: PRINT SPC(46): NEXT
200 FOR T= 0 TO U-1
210 FOR E=27 TO 28-A(T) STEP-1
220 PLOT E, (T+1)*3+3, B$(1)+B$(0)
230 NEXT
240 PLOT 30, (T+1)*3+3, A$(T)
250 NEXT
260 POKE #268,26:PRINT "AGAIN (Y/N)?"
270 GET A$: IF A$="Y" THEN RUN
280 IF A$="N" THEN END ELSE 270
```

The background of the page is a complex, abstract illustration. It features a variety of geometric shapes, including triangles, rectangles, and irregular polygons, some of which are filled with horizontal or diagonal hatching. The shapes are layered and overlap, creating a sense of depth and movement. The color palette is primarily black, white, and shades of gray, with some areas of solid black and others of solid white or light gray. The overall style is reminiscent of mid-20th-century abstract art or graphic design.

Missionaries and Cannibals

This program contains instructions but basically the idea is to get 3 missionaries and 3 cannibals from one side of the river to the other. However only two people are allowed in the boat at any time and if the missionaries are outnumbered by the cannibals they will be eaten! The program will also demonstrate a solution if required.

See page 49 for listing

By C. R. Burnham

Cinderella

See pages 50, 51, 52, 53, 54, 55



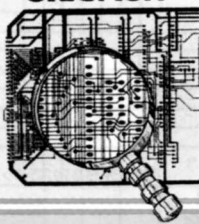
By S. W. Lucas

TANSOFT

ORIC Software

ORIC SOFTWARE

ORIC MON



ORIC-MON

Includes mnemonic assembler/disassembler, block move and verify, data and text entry and cassette and printer handling. Includes manual. £15.0 inc. VAT - 16K or 48K

ORICMUNCH

Munch the Bugmonsters and eat the power pills, clear the maze to reach the high score. £7.95 inc. VAT - 16K or 48K.

ORIC SOFTWARE

ORICMUNCH



TANSOFT

ORIC - BASE



ORICBASE

A complete database system. A sophisticated query language allows you to search and sort your data, keep running totals and print reports. Full manual. £14.50 inc. VAT - 48K.

House of Death

An impressive follow-up to Zodiac, discover the secrets of the haunted house, avoid the witch and the axe murderer and come away with incredible treasures. £9.99 inc. VAT - 48K.

ZODIAC

One of our most successful games. Discover the twelve cunningly disguised signs of the zodiac which will lead you to six treasures. Full of puzzles, puns and misleading clues. £9.99 inc. VAT - 48K.

TANSOFT LANGUAGE MASTERCLASS

If you are going abroad for pleasure or business these courses will give you a basic understanding of the language. Based on the new Gruneberg Link-Word method.

Spanish
Italian
German
French

supplied with program cassette and spoken vocabulary pronunciation cassette. £12.95 each inc. VAT - 48K.



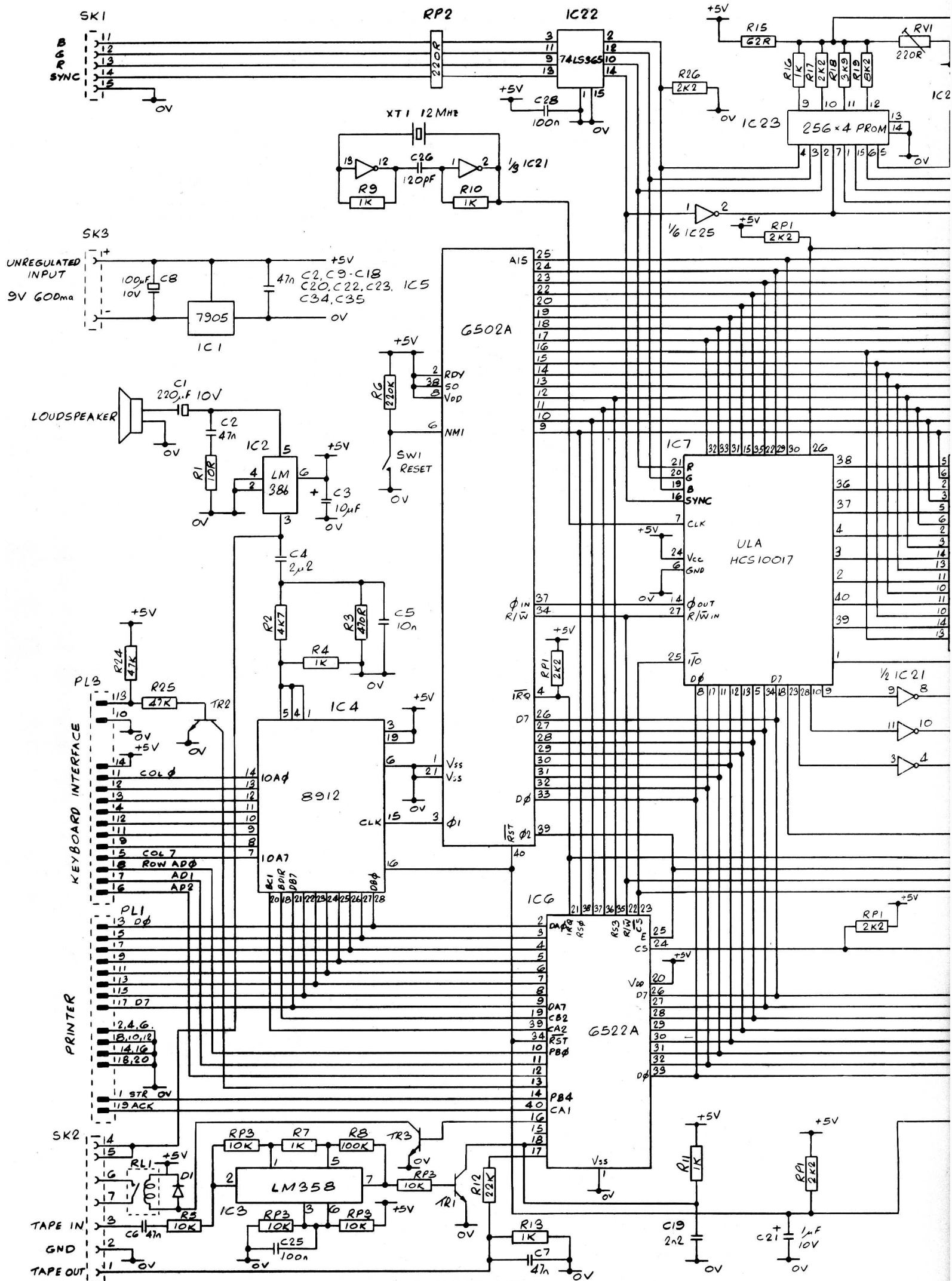
FORTH

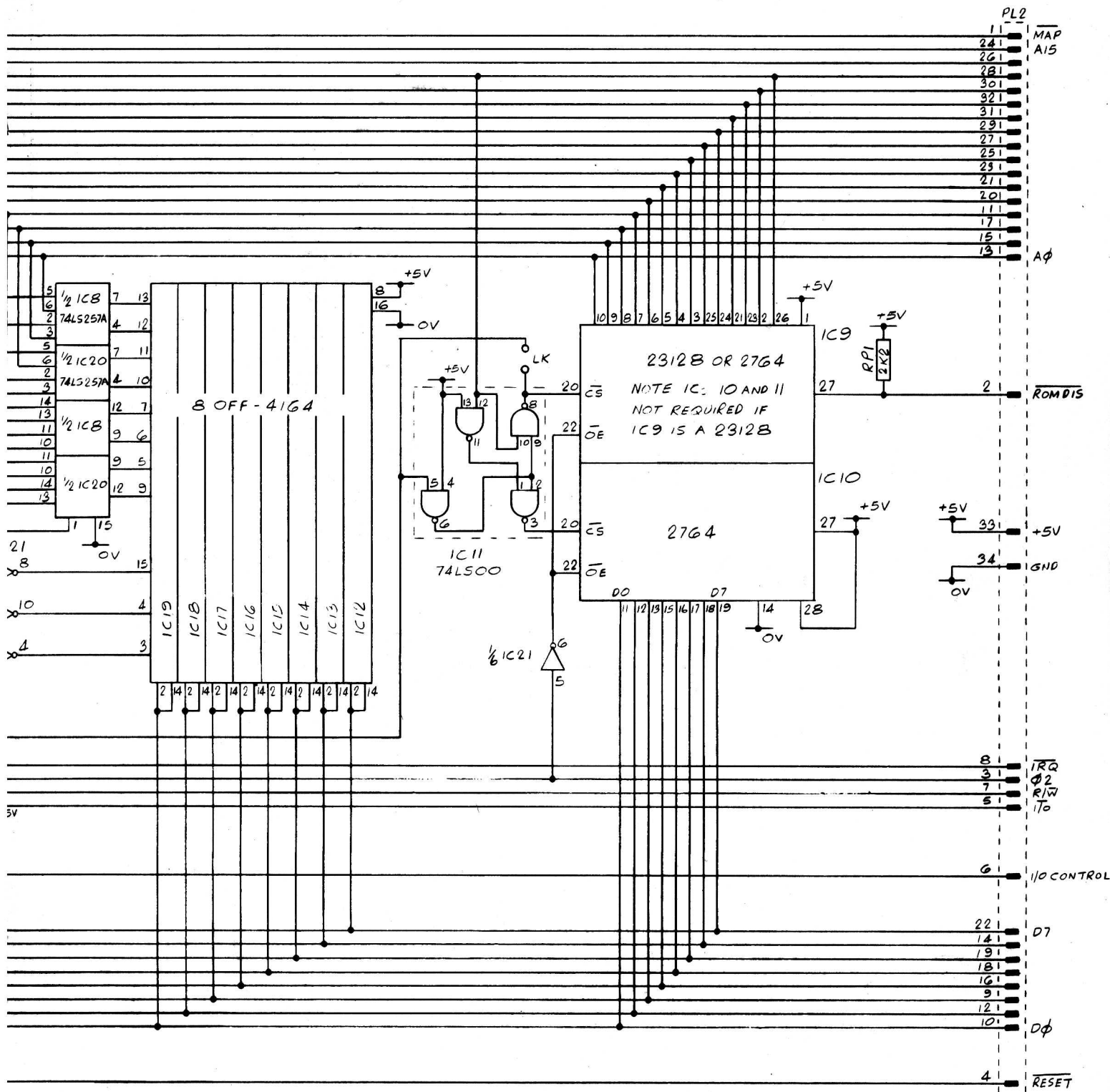
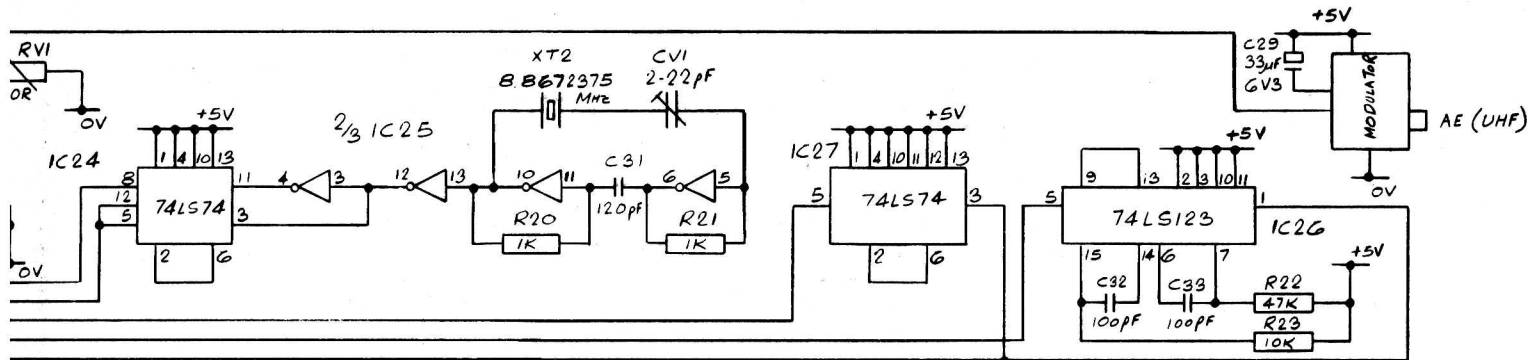
A full implementation of this exciting new computer language. Includes editor and assembler plus language extensions. £15.00 inc. VAT - 48K.



You received your first issue of Oric Owner free with your machine. Why not subscribe now and keep in touch with all the latest developments in the Oric range.

Please send me:		QUANTITY	PRICE	*Delete as appropriate.
TITLE	Oricmon			
TITLE	Forth			
TITLE	Oricbase			
TITLE	House of Death			
TITLE	*French/Spanish/Italian/German			
TITLE	Oricmunch			
TITLE	Oric Owner (6 issues £10.00)			
NAME				
ADDRESS				
I enclose cheque/P.O./Access No.				
Tansoft Ltd., Unit 1 & 2, Techno Park, for £				
Newmarket Road, Cambridge.				
Tel: Order hotline: (02205) 2261.				





ORIC-1 48K

DATE	ISSUE/CHANGE
5	7-12-83 REDRAWN
G	C/N 56 26-9-83

ORIC Software

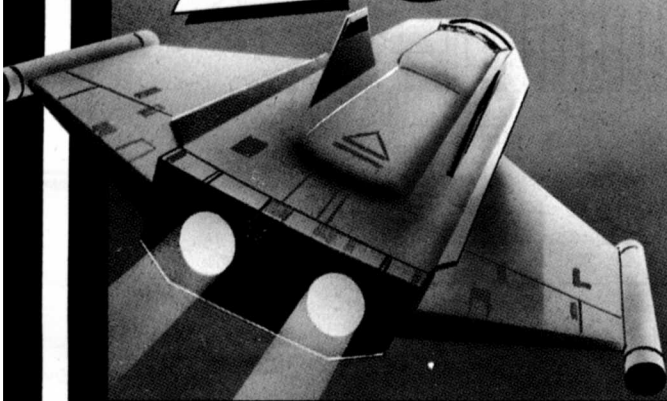
TANSOFT

ultima ZONE

A 100% machine code space trilogy—shoot the Walkons, avoid the bouncing Brunes, find your way through the satellite zone. Requires 48k Oric.

£8.50

inc. V.A.T. post free



The Hobbit

In co-operation
with
Melbourne House.

At last, this best selling adventure is available on the Oric based on J. R. R. Tolkien's book. The Hobbit (included) a full screen text and graphics adventure. Requires 48k Oric.

£14.95

Our software is available from all Oric dealers and most good software suppliers. In case of difficulty please contact us on Teversham (02205) 2261 or write to us at:

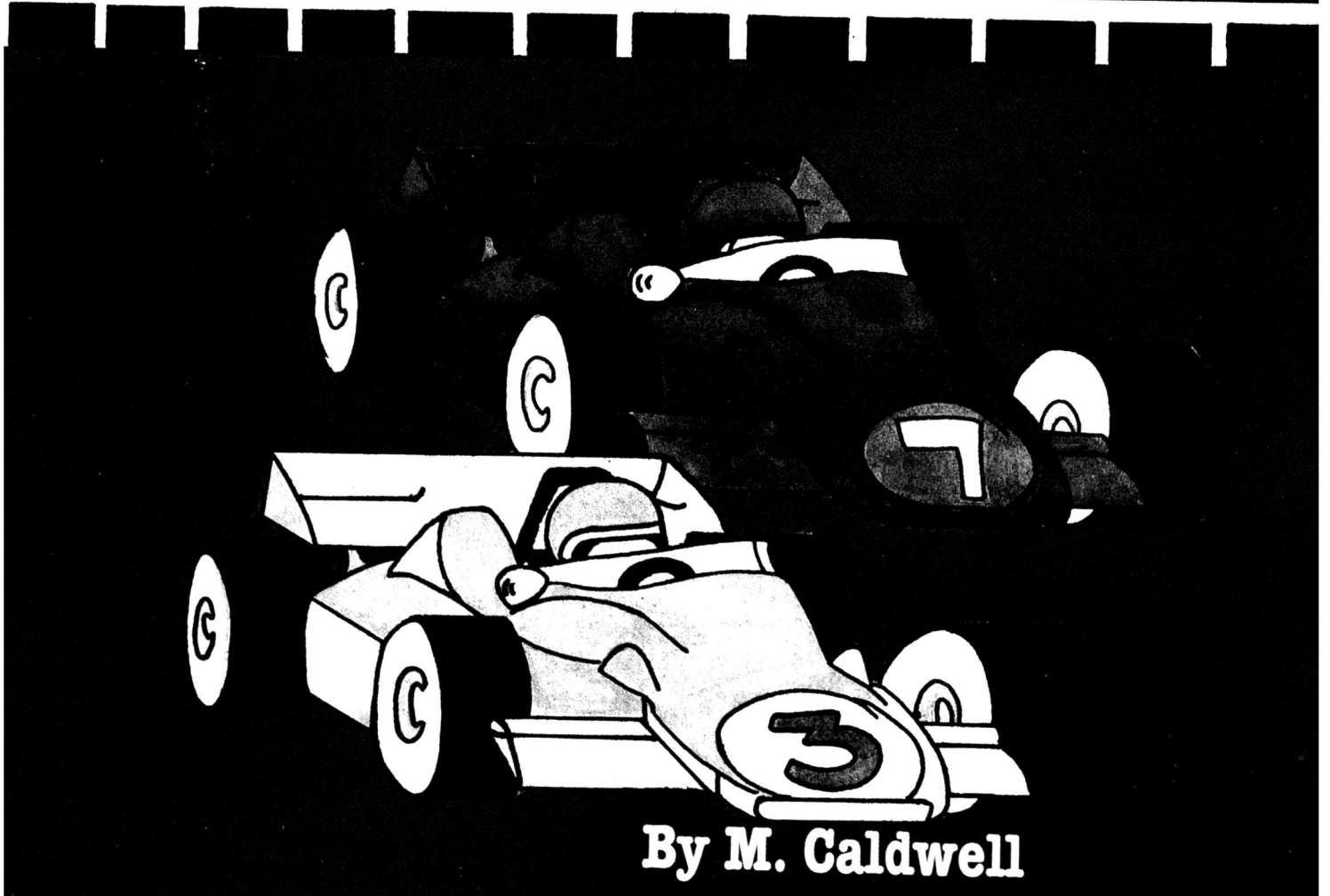
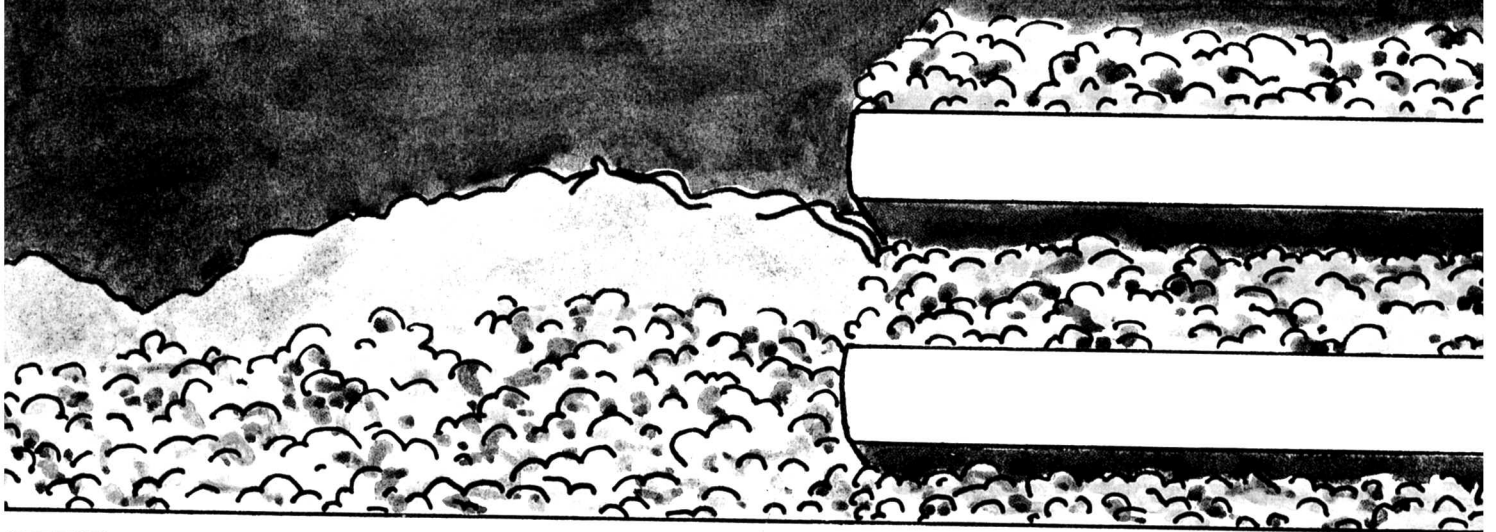
TANSOFT

Unit 1 & 2, Techno Park, Newmarket Road, Cambridge

GRAND *Prix*

You control a car and must do get killed. Although this sounds easy you only have a limited time or fuel to reach the highest mileage. After 400 miles it goes on to the next level.

See page 10 for listing



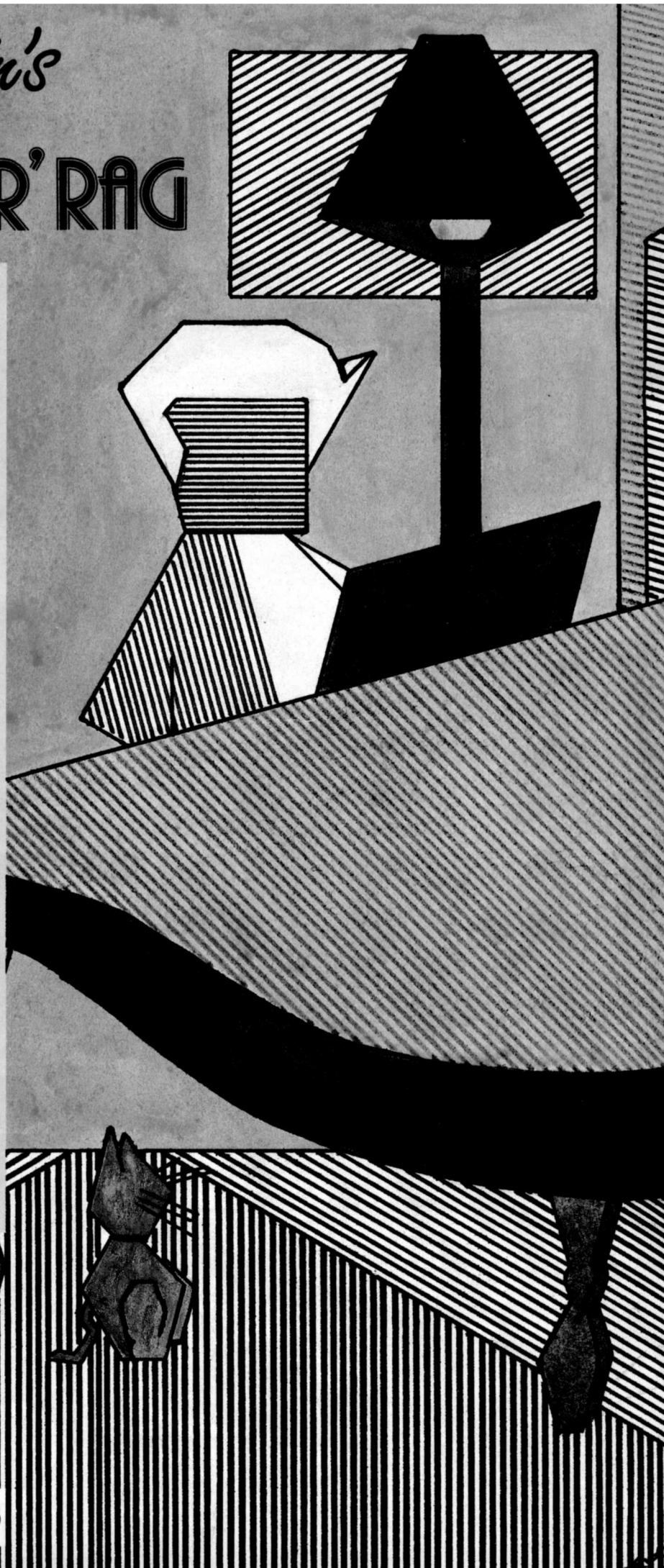
By M. Caldwell

Scott Joplin's 'ENTERTAINER' RAG

This program makes full use of the Oric-1's excellent musical capabilities, by playing this world-famous Ragtime Two-step. The program is written so that typing is kept to a bare minimum, by re-using routines when required rather than typing them all out all over again. The overall effect is helped by using the lower end of the Oric's sound spectrum, which means that the harmonics are somewhat more prominent than higher up, thus creating the impression that there are more than just three voices playing (the original needs 7 voices!). Some may question the use of the MUSIC command throughout rather than DATA. The explanation is quite simple: whilst DATA is perfectly acceptable for one voice only, when three voices are used, the DATA system produces synchronisation troubles, and experience has shown that using the MUSIC statement throughout, although considerably more tedious to type in, should produce by far the best results in the end. Happy typing and listening!

See pages 56, 57 for listing

By J. P. Hibbins



Missionaries and Cannibals Listing

By C. R. Burnham

```

550 IFCR:OTHEN:1,20:CHRE(128);:FORA=1TOCR:1A+31,20;'I';NEXT
555 IFCE='Y'THEN3240:5:READMR:CR:GOT0560
555 PLOT13,21,'6 SPACES':PLOT21,21,'xxxxxx':GOT0620
560 REM **** VALID MOVE FROM RIGHT ****
570 CL=CL+CB:ML=ML+MR:CR=CR+MR:MR=MR+1A+20;'C';NEXT
580 IFCL:OTHEN:1,20:CHRE(135);:FORA=1TOML:1A+1,20;'I';NEXT
590 IFCL:OTHEN:1,20:CHRE(128);:FORA=1TOCL:1A+5,20;'I';NEXT
600 IFMR:OTHEN:1,27,20:CHRE(135);:FORA=1TOMR:1A+27,20;'C';NEXT
610 IFCR:OTHEN:1,31,20:CHRE(128);:FORA=1TOCR:1A+31,20;'I';NEXT
615 PLOT21,21,'6 SPACES':PLOT13,21,'xxxxxx'
617 IFCE='Y'THEN3240
620 REM **** MAKE SURE THEY'RE NOT OUTNUMBERED ****
630 IFML>OANDCL>MLTHENZAP:PLOT1,1,'YOU FOOL,YOU'VE FED THE CANNIBALS'
640 IFMR>OANDCL>MRTHENZAP:PLOT1,1,'YOU FOOL,YOU'VE FED THE CANNIBALS'
650 IF (ML>OANDCL>ML)OR(MR>OANDCR>MR)THENWAIT400:GOT03100
660 REM **** GAME OVER YET ? ****
670 IFCR+MR<6THEN320?'NOT YET'
680 PING:PLOT3,1,'WOW I, SO WHO CARES IF YOU DID GET'
690 PLOT3,3,'THEM ACROSS ANYWAY ????:WAIT400:GOT03100
1900 REM **** INVALID REPLIES ****
1910 ZAP:PLOT1,1,'HOW AM I SUPPOSED TO DO THAT ?':WAIT150
1920 ZAP:PLOT1,1,'30 SPACES':RETURN
1930 ZAP:PLOT1,1,'LOOK AT THE MISSIONARIES AGAIN':WAIT150
1940 ZAP:PLOT1,1,'30 SPACES':RETURN
1950 ZAP:PLOT1,1,'LOOK AT THE CANNIBALS AGAIN !?':WAIT150
1960 ZAP:PLOT1,1,'30 SPACES':RETURN
2000 REM **** INSTRUCTIONS ****
2010 CLS:PRINT:PRINT
2020 PRINT'3 MISSIONARIES ('CHRE(135)')<<<'CHRE(128);
2030 PRINT' ) & 3 CANNIBALS (111)';
2040 PRINT' ARE ON THE LEFT BANK';
2050 PRINT'OF A RIVER THAT IS TO DANGEROUS TO SWIM.';
2060 PRINT'ALL 6 WISH TO CROSS OVER TO THE RIGHT BANK.'
2070 PRINT'TO DO SO THEY HAVE A SMALL BOAT';
2080 PRINT' WHICH WILL CARRY 2 PEOPLE AT A TIME.'
2090 PRINT'TO ENSURE THAT THEY WILL NOT '
3000 PRINT'BE EATEN';
3005 PRINT'THEY MUST NEVER BE OUTNUMBERED.'
3010 PRINT:PRINT:PRINT
3020 PRINTCHRE(135) HOW CAN ALL 6 CROSS SAFELY?:WAIT2000
3030 PRINT:PRINT:PRINT'DO YOU WANT TO CHEAT AND SEE THE'
3040 PRINT'SOLUTION ?':GETA:IFA='Y'THENCE='Y':GOSUB3200
3050 RETURN
3100 REM **** ANOTHER GAME ? ****
3110 CLS:PLOT10,10,'GAME OVER - DO YOU'
3120 PLOT10,12,'WANT ANOTHER ONE ?':GETA:
3130 IFA='Y'THENRUN230ELSECLS:PRINTCHRE(6)CHRE(17):NEW
3200 REM **** DEMO OF SOLUTION ! ****
3210 CE='Y':GOT0230
3230 MB=0:CB=2:GOT0500
3240 WAIT75:READMB:CB:IFMB+CB>0THENGOT0500ELSERETURN
3250 DATA:1,2,,1,2,,1,1,2,,1,2,,1,2,,1,2,,SECRET DATA !

```

OF RECORDS DUMPED 107

Cinderella

In this adventure, you must take on the roll of Cinderella. The game loosely follows the plot of the fairy tale and like all adventure games part of the fun in playing is to discover the solution for yourselves. I would suggest that you do NOT read the hints for solution of the game UNLESS you are completely stuck.

The game starts with you (Cinderella) in the kitchen near to the fireplace. In a few hours time, there is to be a Ball in the Royal Palace to which everybody except you has been invited. Since your mother died a few years ago, you have lived in a large house a few miles from the Royal Palace with your elderly father and two UGLY sisters. The Ball is to be held to celebrate the 21st birthday of Prince Charming and you would dearly love to go to the ball to meet the prince.

To solve the game completely, you must try to get the prince to marry you and the final moves of the game will involve you in getting married to the prince. BEWARE, if you lose the game, he will marry one of your UGLY sisters!

You must give the computer instructions as to what you would like to do in the form of one or two words such as GET CINDERS, EMPTY DUSTBIN etc.

There are 72 locations to visit in this game.

WORDS UNDERSTOOD

GO, N,S,E,W, UP, DOWN, IN, OUT, EMPTY, DRINK, EAT, LOOK, SCORE, RUB, DRINK, HELP, RUN, SAY, SPEAK, WEAR, TRY ON, PRAY, CRY, ASK, READ, PHONE, BUY, INVENTORY, ROW, SAIL, PUSH, WASH UP, SWEEP FLOOR, DANCE, RIDE, WAIT, JUMP, DIVE, GET, TAKE, GRAB, DROP, LEAVE, PUT, OPEN, UNLOCK, KISS, GIVE, EXAMINE, SEARCH, KILL, SHOOT, ATTACK, DESTROY, SWIM

GENERAL RULES

1. In each location HELP will often give you a hint.
2. Whenever you are sad and CRY then your fairy godmother will be thinking about you and will often appear to give you assistance. (PRAYING is

another way of calling for your fairy godmother).

HINTS

Read these only if you are really stuck for what to do!!!

1. You must first find the pumpkin, six mice, two lizards and one rat and take these to the back yard. – try CRY at the beginning.
2. The pumpkin is in the vegetable plot but you will need the spade to dig it up! This is to be found inside the garden shed.
3. The lizards are to be found inside the cucumber frame. Type HELP when you are near to the cucumber frame.
4. The mice are in the mouse trap in the secret room off the oak panelled library. Type HELP in the library.
5. The rat is in the trap in the cellar.
6. If your ugly sisters demand that you do something, IT is important that you do as they ask.
7. Do not swear at the computer!
8. When you have taken these items to the back yard, call for your fairy godmother TWICE!!
9. She will tell you to get some shoes to wear.
10. When you have found these (do not get them before she has told you to do so!), call for your fairy godmother again!

11. If you are stuck inside a glass coach or on horseback just WAIT around awhile!
12. When you get to the palace go to the ball and find the prince.
13. When you find the prince DANCE with him (or you could even KISS him!)
14. If it is close to midnight, BEWARE!! Your clothes will

turn to rags if you hang around so RUN for your life.

15. If the prince's servants come along asking 'to whom the slipper belongs' why not try it on!

16. ONLY when the fairy godmother tells you to do so should you get the WEDDING DRESS or your sister will hit you and you will lose the game.

17. You will also lose if you cut

yourself to death with the cucumber frame!

18. When you have got the wedding dress, you must prepare for the wedding by WEARING the dress. Your fairy godmother should then appear and wave her wand.

19. Once you have reached the church, go in and you should then be able to marry your prince.

Cinderella Listing

```

10 REM** CINDERELLA ---an adventure game for the 48K Oric1 **
20 REM ** <C> October 1983 **
30 REM ** S.W.Lucas **
35 PRINTCHR$(17):REM TURNS OFF CURSOR
37 PAPER0:INK6
40 TEXT:GOSUB3000:REM TITLES
100 P%:=9:R%:=0:SA:=0:SB:=0:SC:=0:SD:=0:SE:=0:SF:=0:SG:=0:SH:=0:SI:=0:
110 DIMS$(103,4),X$(35),N$(33),G$(103),V$(30),B$(30,1),N$(33)
130 GOSUB800:REM READ DATA INTO THE ARRAYS
140 CLS:REPEAT
150 GOSUB2280
160 IF P%=72 THEN GOSUB2520
170 IFM%:=19AND(P%:=10ORP%:=33) THENGOSUB2600
171 IFP%:=33THENS$(1,4):=33:G$(1):="by the open door"
180 PRINTCHR$(131):"I am :-"
181 PRINTG$(P%):AS$=":"
184 IFS$(P%,1)<>0 THEN AS$="North "
190 IFS$(P%,2)>0 ANDLEN(AS$)>0 THENAS$=AS$+"South "
195 IFS$(P%,2)>0 ANDLEN(AS$)>0 THENAS$="South "
200 IFS$(P%,3)>0 ANDLEN(AS$)>0 THENAS$=AS$+"East "
205 IFS$(P%,3)>0 ANDLEN(AS$)>0 THENAS$="East "
210 IFS$(P%,4)>0 ANDLEN(AS$)>0 THENAS$=AS$+"West "
211 IFP%:=18ANDX<>1 THENPRINT"One of my UGLY sisters rushes in and"
220 IFP%:=20ANDX<>1 THENPRINT"tells me to wash up NOW"
221 IFP%:=20ANDX<>1 THENPRINT"One of my UGLY sisters insists that I"
221 IFP%:=20ANDX<>1 THENPRINT"sweep the floor. She will not let me"
230 go North"
230 IFS$(P%,4)>0 ANDLEN(AS$)>0 THENAS$=AS$+"West "
235 IFS$(P%,4)>0 ANDLEN(AS$)>0 THENAS$="West "
240 IFS$(P%,1)>0 ANDS$(P%,2)=OANDS$(P%,3)=OANDS$(P%,4)=O THENAS$="nowh"
240 ere ?"
260 PRINT:PRINTCHR$(130):"I can go :- ":AS$:PRINT:PRINT
270 E:=O:FORI:=1TO20:PF%:=O:IFB$(I,1)=P% THENPF%:=1
280 NEXTI:GOTO320
290 IFE=O THENPRINTCHR$(134):"That looks like :-"
310 PRINTG$(I):E=E+1:GOTO290
320 PRINT:PRINTCHR$(133):"What do I do now ":INPUTZ$
322 B$=LEFT$(Z$,2):C$=LEFT$(Z$,3)
330 A$=FRE(""):REM TO COLLECT GARBAGE & PREVENT THE CHARACTER SET BE
330 ING ERASED
335 CLS:PRINTCHR$(134)"Hold on a second!":PING
340 K=0
350 IFC$="LOO" THENK=1
360 IFC$="SCD" THENPRINTCHR$(131)"This isn't a game you know":K=1
370 IFC$="EAT" THENPRINTCHR$(129):"I'm not hungry at the moment":K=1
375 IFC$="RUB" THENPRINTCHR$(MUSIC131):"I can't do that at the moment"
380 IFC$="DRI" THENPRINTCHR$(130)"I only drink milk & I can't see an"
390 IFC$="HEL" THENK=1:GOSUB2350
400 IFC$="RU" THEN GOSUB2570
410 IFC$="SAY" ORC$="SPE" THENPRINTCHR$(129):"I talk but nobody liste"
420 IFC$="WEA" ORC$="TRY" THENGOSUB2630
430 IFC$="PRA" ORC$="CRY" ORC$="ASK" THENK=1:GOSUB2130
440 IF (B$="N" ORLEFT$(Z$,4)="GO ") ANDS$(P%,1)
450 IF (B$="S" ORLEFT$(Z$,4)="GO S") ANDS$(P%,2)<>0 THENP%:=S$(P%,2):K=1
460 IF (B$="E" ORLEFT$(Z$,4)="GO E") ANDS$(P%,3)<>0 THENP%:=S$(P%,3):K=1
470 IF (B$="W" ORLEFT$(Z$,4)="GO W") ANDS$(P%,4)<>0 THENP%:=S$(P%,4):K=1
480 IFB$="RE" THENPRINT"I'm sorry I forgot my glasses!":K=1
490 IF (B$="S" ORB$="N" ORB$="E" ORB$="W") ANDK<>1 THENPRINT"I can't!":K=
500 IFC$="PHO" THENPRINT"I'm not E.T. You know! I need a phone":K=1
510 IFC$="FU" ORC$="PIS" THENPRINTCHR$(129)"HOW DARE YOU SPEAK TO ME"
520 IFC$="BUY" THENPRINT"MY sisters don't give me money!":K=1
530 IFC$="INV" THENGOSUB1700
540 IFC$="ROW" ORC$="SAI" THENPRINT"I'd need a boat to do that!":K=1

```

```

550 IFR$="UP"ORZ$="G UP"THEN GOSUB 1010
560 IFC$="DOW"ORZ$="GO DOW"THEN GOSUB 1040
570 IFC$="OUT"ORZ$="GO OUT"THENGOSUB 1070
580 IFC$="I"ORZ$="GO IN"THENGOSUB 1100
590 IFC$="PUS"THENGOSUB 2420
600 IFC$="WAS"THENGOSUB 2100
610 IFC$="DAN"THENGOSUB 2500
620 IFC$="RID"ORZ$="GO RIDE"ORZ$="GO COACH"THENPRINT "not yet":K=1
630 IFC$="WAI"THENGOSUB 2460
640 IFC$="JUM"ORC$="DIV"THENK=1:GOSUB 2440
650 IFC$="GET"ORC$="TAK"ORC$="GRA"THENK=1:GOSUB 1310
660 IFC$="DRQ"ORC$="LEW"ORC$="PUT"THENGOSUB 1580
542 IFC$="SWE"THENGOSUB 2240
680 IFC$="DPE"THENPRINTCHR$(129)"I'm afraid I can't do that just yet":K=1
690 IFC$="UNL"THENGOSUB 1790
700 IFC$="KIS"THENK=1:GOSUB 1820
710 IFC$="GIV"THENK=1:GOSUB 1850
720 IFC$="EXA"THENPRINT "I can't see anything special":K=1
730 IFC$="SEA"THENGOSUB 2220
740 IFC$="KIL"ORC$="SHO"ORC$="HIT"ORC$="ATT"ORC$="DES"THENK=1:GOSUB 1870
745 IFC$="EMP"THENPRINT "I don't think that would do much good":K=1
750 IFC$="SWI"THENK=1:PRINT "just give me a direction & I'll swim if I have to"
760 IFK<>1THENPRINTCHR$(135)"I'm sorry I don't understand that"
770 UNTILPZ=71:REM POSITION TO WIN
780 GOTO850
790 END
900 FORH=1TO72:READC$(H):FORD=1TO4:READS$(H,D):NEXTD,H
810 FORH=1TO23:READB$(H),BZ(H,1):NEXT:FORM=1TO33:READN$(H),NX(H):NEXT
830 RETURN
840 REM ROUTINE FOR WINNING!
850 CLS
860 FORX=1TO2
870 PLOT1,X,14:PLOT2,X,X+1:PLOT10,X,"W E L L D O N E"
880 NEXT
990 PLOT1,10,1:PLOT2,10,"You have solved this adventure"
900 PLOT1,15,3:PLOT2,15,"I marry the prince and live happily"
905 PLOT1,16,3:PLOT2,16,"ever after!"
910 FORX=1TO5:FORY=1TO220 STEP 20
920 SOUND1,Y,0:PLAY1,0,3,2000
925 WAIT10
930 NEXT Y,X
940 PLAY0,0,0,0
950 PLOT1,24,2:PLOT2,24,"Do you want another game <Y>es / <N>o"
960 REPEAT
970 GETS$
975 UNTIL (S$="Y"OR S$="N")
980 IFS$="Y"THENRUN
990 END
1000 REM ROUTINE TO GO UP
1010 K=1
1015 IFPZ=2THENPRINT "O.K.":PZ=23:RETURN
1016 IFPZ=13THENPRINT "It's too tall for a poor defenceless person!":RETURN

```

```

1018 IFPZ=28THENPRINT "I climb the cellar steps!":PZ=21:
1020 PRINT "I can't do that at the moment!"
1022
1030 REM ROUTINE TO GO DOWN !
1040 K=1
1044 IFPZ=23THENPRINT "I go down the stairs!":PZ=2:
1046 IFPZ=21THENPRINT "O.K.":PZ=28:RETURN
1050 PRINT "Don't be silly now!!!":RETURN
1060 REM ROUTINE TO GO OUT
1070 K=1
1072 IFPZ=11THENPRINT "O.K.":PZ=12:RETURN
1073 IFPZ=64THENPRINT "O.K.":PZ=63:
1075 IFPZ=60THENPRINT "O.K.":PZ=59:RETURN
1077 IFPZ=1THENPRINT "I can't find the key!":RETURN
1080 PRINT "I'm sorry I can't do that here!":K=1:RETURN
1090 REM ROUTINE TO GO IN
1100 K=1
1101 DATA 5,5,8,5,8,4,8,5,7,5,6,4,7,3,8,6,9,4,9,5,6,3,8,7,6,5,4,7,4,8,2,9,4,5
1104 IFPZ=12THENPZ=11:PRINT "O.K.":RETURN
1105 IFPZ=63THENPZ=64:PRINT "O.K.":
1106 IFPZ=59THENPZ=60:PRINT "O.K.":RETURN
1108 IFPZ=61THENPRINT "It's locked":RETURN
1109 PRINT "Don't be DAFT!!!":ZAP:RETURN
1110 DATA by a large Oak door. It is locked.,0,0,2,0
1111 DATA in a hallway. The staircase lead DEEK up from here.,0,0,3,1
1120 DATA in the hall,4,19,8,2,in a large oak panelled library,0,3,0,0
1121 DATA in a secret passage,0,4,6,0,in a musty secret room,0,0,0,5
1130 DATA at the western end of the dining room,0,8,10,0
1131 DATA in a large kitchen,7,18,9,3,by a large old fashioned fireplace
1132 DATA 0,17,0,8,by the dining table,0,0,0,7
1140 DATA in ride the garden shed. It is full of old plant pots!,0,0,0,0
1141 DATA by a garden shed. The door is open.,0,13,0,0
1142 DATA in a large orchard full of apple trees,12,14,0,0
1150 DATA on a large well kept lawn,13,0,0,15,in my father's back yard
1151 DATA 0,16,14,17,in the vegetable patch,15,0,0,0
1152 DATA by the open back door of my father's house,9,0,15,0
1160 DATA by the kitchen sink,0,0,0,0,in a large lounge,3,20,0,0
1161 DATA by the broom cupboard door.,0,21,0,0
1162 DATA inside the broom cupboard. I can go down into the cellar from here
1163 DATA 20,0,0,0
1170 DATA in the large bedroom where my sisters sleep.,0,23,0,0
1171 DATA on the landing,22,27,25,24,in my father's bedroom,0,0,23,0
1172 DATA in the bathroom,0,26,0,23
1180 DATA in the toilet,25,0,0,0,in the boxroom. It is full of boxes,23,0,0,0
1181 DATA in the cellar. The steps lead up from here,0,0,29,0
1190 DATA at the far end of the cellar,0,0,0,28
1191 DATA on a main road. The way North is blocked by a fallen tree
1192 DATA 0,31,0,0,on the road,30,32,0,0

```

Cinderella Listing

Software

Cinderella Listing

```

1193 DATAoutside my garden gate,31,34,33,0
1194 DATAoutside my front door. It is open,0,0,1,32
1200 DATAon the main road leading orth/South,32,35,0,0
1201 DATAat the entrance to the castle grounds,34,56,36,0
1202 DATAoutside a large gothic castle,0,0,37,35
1210 DATAon the castle steps.,0,0,38,36
1211 DATAin a huge entrance hall full of      ancient sepulchres,4
0,39,49,37
1212 DATAin a room full of cobwebs,38,0,0,0
1220 DATAin a long narrow North/South corridor.Paintings line the w
alls
1221 DATA45,38,41,0,in a large banquet hall.,44,0,0,40
1222 DATAin the palace kitchens,43,0,0,0
1223 DATAin a narrow passage,0,42,0,44
1224 DATAin the banqueting hall. There is      nobody here!,48,41,4
3,0
1230 DATAby a window looking out over the lawns,47,40,0,46
1231 DATAon a vast lawn,0,0,45,0
1232 DATAat LEFT$he end of the passage. There is a door here but it
is locked.
1233 DATA0,45,0,0,in a small lounge,0,44,0,0
1240 DATAat the entrance to the ballroom,0,0,50,38
1241 DATAin the ballroom. It is very crowded.,0,51,0,49
1242 DATAin a crowd of people at one side of   the ballroom..50,57.
52,0
1243 DATAon the floor of the dance. A young manasks me to dance,0,5
4,0,51
1244 DATAby a table,51,0,54,0,in a crowd of people,52,55,0,53
1250 DATAin the ballroom,54,0,0,0
1251 DATAin the village square,35,63,57,0
1252 DATAin a narrow cobbled street.,0,0,58,56
1260 DATAin a small courtyard,0,59,0,57,outside a cafe,58,0,0,0
1261 DATAinside the cafe,59,0,0,0
1262 DATAby the village blacksmith's. It is   closed.,0,0,0,40
1270 DATAat the South end of the village square,60,1,61,0
1271 DATAoutside the village shop,56,62,0,0
1272 DATAinside the village shop,0,0,63,0
1273 DATAriding inside a beautiful glass coach,0,0,0,0
1274 DATAin a stationary coach,0,0,0,0,riding on a horse,0,0,0,0
1280 DATAoutside the village church,69,0,0,0
1281 DATAin the entrance to the village church,70,68,0,0
1282 DATAwalking down the aisle. The organ is playing the wedding
g march**
1283 DATA71,69,0,0,at the altar.Prince charming is here.,0,70,0,0
1284 DATA dancing with a handsome prince on a   very crowded danbe f
loor.
1285 DATA0,0,0,0
1300 REM ROUTINE FOR GET
1310 GOSUB 1500
1315 K=1
1320 IFLX<>1THENRETURN
1330 EX=0
1340 FORH=1TO23
1344 IFBZ(H,1)=PZ ANDBZ(NZ(R),1)=PZ THENEX=1
1350 NEXT
1360 IFEZ=0THENPRINTCHR$(129)"I can't see it here!":K=1:RETURN
1370 IFR=13THENSA=1

```

```

1380 IFR=5THENX$(1)="I GRAB HOLD OF THEM & BURN TO DEATH!":GOSUB189
0
1390 IFR=16ORR=17ORR=31ORR=32ORR=33THENPRINTCHR$(135)"Don't be STUP
ID!":RETURN
1400 IFR=10ANDSB<>1THENPRINTCHR$(129)"I need a spade to dig that":
RETURN
1410 IFR=6THENSB=1
1420 IFR=23ORR=24THENX$(1)="The glass breaks and cuts me to death!":
GOSUB1890
1430 IFR=14ORR=15THENGOSUB2680
1440 IF(R=18ORR=19ORR=20)ANDBZ=5THENPRINT"My fairy godmother reappe
ars and"
1441 IF(R=18ORR=19ORR=20)ANDBZ=5THENPRINT"waves her MAGIC WAND!":PZ
=65
1444 IF(R=18ORR=19ORR=20)ANDBZ=5THENPRINT"I am wearing a beautiful
golden dress"
1445 IF(R=18ORR=19ORR=20)ANDBZ=5THENPRINT"and glass slippers"
1446 IF(R=18ORR=19ORR=20)ANDBZ=5THENPRINT"She says 'be sure to retu
rn by 12pm'"
1447 IF(R=18ORR=19ORR=20)ANDBZ=5THENK=1:
1450 EX=0
1455 FORD=1TO3
1457 IFV$(D)="" THEN V$(D)=6$(NZ(R)):EX=1:D=5
1460 NEXT
1470 IFEZ=0THENPRINT"I'm sorry I just can't carry any more!":K=1:RE
TURN
1480 BZ(NZ(R),1)=0:K=1:RETURN
1490 REM CHECK ROUTINE
1500 L$=""
1505 FORH=1TOLEN(Z$)
1510 IF MID$(Z$,H,1)="" THENL$=RIGHT$(Z$, (LEN(Z$)-H)):H=H+40
1520 NEXT
1530 R=0:LZ=0
1535 IFLN(L$)<2THENK=1:RETURN
1540 FORH=1TO27:IFLEFT$(N$(H),LEN(L$))=L$ THENLZ=1:R=H
1550 NEXT
1560 K=1:RETURN
1570 REM ROUTINE FOR DROP
1580 GOSUB1500
1590 IFLX<>1THENPRINT"I can't see a :-":L$=K=1:RETURN
1600 K=1:EX=0
1610 FORD=1TO3
1620 IFV$(D)=6$(NZ(R)) THEN V$(D)="" :EX=1
1630 NEXT
1640 IFEZ<>1 THENPRINT"I haven't got it!!!!":RETURN
1650 BZ(NZ(R),1)=PZ
1660 CLS
1670 IFR=13THENSA=0
1680 K=1:RETURN
1690 REM IVENTORY ROUTINE
1700 PRINTCHR$(129);"I have :-"
1710 FZ=0
1720 FORH=1TO3
1730 IFV$(H)<>""THENPRINTV$(H):FZ=1
1740 NEXT
1750 IFFZ=0THENPRINTCHR$(131);"Nothing at all!"
1760 PRINT

```

```

1770 K=1:RETURN
1790 K=1:IFPZ<>1THENPRINT"Don't be a SILLY BILLY!
1800 PRINT"I need the $KEY$ to unlock the door":RETURN
1810 REM KISS ROUTINE
1820 IFPZ=55THENPZ=72:PRINT"The prince takes me for a dance!":K=1:R
ETURN
1830 PRINT"There is nobody here I feel like kissing!":K=1:RETURN
1840 REM GIVE ROUTINE
1850 K=1:PRINT"There is nothing I'd like to give!":RETURN
1860 REM KILL ROUTINE
1870 IFPZ<>103THENPRINT"Not here and now!":K=1:RETURN
1880 REM LOSE ROUTINE
1890 CLS:FORX=1TO2
1900 PLOT1,X,14:PLOT2,X,X+1:PLOT3,X,"That's done it !!!!!!!!!!!!!!!
"
1910 NEXT
1920 RESTORE
1925 PLOT1,10,1:PLOT1,10,X$(1)
1930 PLOT1,15,3:PLOT2,15,"The handsome prince has married my"
1934 PLOT1,17,2:PLOT15,17,"UGLY sister"
1936 PLOT1,22,1:PLOT2,22,"How could he do this to me ??????"
1812 FORT=1TO34:READX
1945 MUSIC1,3,X,0
1947 PLAY1,0,4,500
1950 WAIT20
1955 NEXT
1956 PLAY0,0,0,0
1957 SHOOT:WAIT20:SHOOT:WAIT10:SHOOT
1960 PLOT1,24,6
1970 PLOT2,24,"Do you want another gamd <Y/N>?"
1980 REPEAT
1982 GETA$
1990 UNTIL(A$="Y"OR A$="N")
2000 IFA$="Y"THENPRINTCHR$(17):RUN:REM TURN CURSOR OFF
2010 CLS:PLOT1,15,1:PLOT2,15,"Thank you for playing!!!"
2020 PLOT1,25,3:PLOT10,25,"GOODBYE!!!!!"
2025 END
2040 DATAa mouse trap.There are 6 mice in it,6
2041 DATAsome cinders,9,a spade,11,an apple tree,13,"",15,a dustbin
,15
2042 DATAa large pumpkin,16,some dirty pots,18,a broom,20
2044 DATAa beautiful wedding dress **,22,one of my ugly sisters,2
2
2050 DATAa pair of old shoes,27,a rat trap with one rat in it,29
2052 DATAa cucumber frame,14,"",14,some food,42,some flowers,46
2053 DATAa tray of drinks,53,a receptionist,60,Prince Charming,55
2054 DATathe prince,71
2060 DATAmy ugly sisters,53,"",9
2070 DATAMOUSE,1,MICE,1,TRAP,1,HRSE,1,CINDERS,2,SPADE,3,APPLE,4,TRE
E,4
2072 DATADUSTBIN,6,PUMPKIN,7,CACH,7,POTS,6,BROOM,9,WEDDING,10,DRESS
,10
2080 DATAUGLY,11,SISTERS,11,SHOES,12,SLIPPER,12,GLASS,12,RAT,13,TRA
P,13
2082 DATACUCUMBER,14,FRAME,14,LIZARDS,15,FOOTMEN,15,FOOD,16
2085 DATAFLOWER,17,TRAY,18,DRINKS,18,RECEPTIONIST,19,PRINCE,20,SIST
ERS,22

```

```

2086 DATAASLIPPER,5,GLASS,5
2090 REM WASH UP ROUTINE
2100 K=1
2102 IFPZ<>18THENPRINT"NOT NOW!!!!":RETURN
2110 PRINTCHR$(129)"I wash all the dishes and my UGLY"
2111 PRINTCHR$(130)"sister goes away!":AZ=1:SZ(18,1)=8:G$(8)="some
clean pots"
2120 REM SUBROUTINE FOR PRAYING/CRYING!
2130 K=1
2135 IFPZ<>22THENPRINTCHR$(129)"Nothing happens!":RETURN
2140 ZAP:WAIT20:PING:WAIT20:ZAP
2150 IFBZ=4THEN2180 ELSEIF BZ=5 THEN2200
2160 PRINTCHR$(131)"My fairy godmother appears and tells"
2164 PRINTCHR$(131)"me to go and find:--"
2170 PRINT"six mice,one rat,two lizards and "
2175 PRINT"the biggest PUMPKIN I can find"
2177 PRINTCHR$(129)"Bring these to the back yard' she":PRINTCHR$(1
29)"says"
2179 RETURN
2180 PRINTCHR$(129)"MY FAIRY GODMOTHER REAPPEARS!":WAIT10:ZAP
2190 PRINT"She waves her magic wand & something magical happens!"
2191 G$(1)="six magnificent white horses"
2192 G$(7)="a beautiful glass coach"
2193 G$(13)="a handsome footman"
2194 G$(15)="two coachmen"
2195 S=FRE("")
2196 AB=1:RETURN
2200 PRINT"My fairy godmother reappears & says ";CHR$(129)"you are
doing well!"
2202 PRINTCHR$(131); "now find some";CHR$(129)"SHOES";CHR$(131)"to w
ear!"
2205 RETURN
2210 REM SEARCH ROUTINE
2220 K=1:IFPZ<>14THENPRINT"I h'ven't found anything!":
2225 PRINTCHR$(129)"I can see something run out of the ":PRINTCHR$(
129)"frame."
2227 G$(15)="Two small lizards":RETURN
2230 REM ROUTINE FOR SWEEP
2240 K=1:IFPZ<>20THENPRINT"NOT HERE!"GRETURN
2250 IFA<>1THENPRINT"I need something to do that with!":RETURN
2260 PRINTCHR$(129)"O.K. my UGLY sister likes that":SZ(20,1)=19
2262 CX=1:RETURN
2270 REM ** SCORE ROUTINE
2280 BZ=0:IFBZ(13,1)=15THEN BZ=BZ+1
2290 IFBZ(15,1)=15 THEN BZ=BZ+1
2300 IFBZ(1,1)=15 THEN BZ=BZ+1
2310 IFBZ(7,1)=15 THEN BZ=BZ+1
2320 IF AB=1THENBZ=BZ+1
2330 RETURN
2340 REM ROUTINE FOR HELP STATEMENTS
2350 K=1
2355 IFPZ=4THENPRINTCHR$(131)"MOST OLD HOUSES HAVE SECRET PANELS!"
2356 IFPZ=4THENPRINTCHR$(129)"Why not try PUSHing the PANEL":RETURN
2360 IFPZ=55THENPRINTCHR$(129);"I'd have a dance if I were you!":RE
TURN
2363 IFPZ=72THENPRINTCHR$(129);"I'd RUN home if I were you!":RETURN
2370 IFMZ=19THENPRINTCHR$(129);"I'd go home!":RETURN

```

Cinderella Listing

Software

Cinderella Listing

```

2380 IFPZ=14THENPRINTCHR$(131)"SEARCH and you shall find":RETURN
2390 IFPZ=65ORPZ=66ORPZ=67THENPRINTCHR$(129);"If I were you I'd WAI
T around"
2392 IFPZ=65ORPZ=66ORPZ=67THENRETURN

```

```

2400 PRINTCHR$(130)"I'm sorry I don't have a clue!"
2401 PRINTCHR$(129);"BUT a good CRY is often useful!!"
2402 PRINTCHR$(130);"and I often get help when I say my":PRINT" pra
yers!"

```

```
2404 RETURN
```

```
2410 REM PUSH ROUTINE
```

```
2420 K=1
```

```
2422 IFPZ<>4THENPRINTCHR$(129);"Not at the moment!":
```

```
2424 PRINTCHR$(129)"A SECRET p'nel slides to one side":S%(4,1)=5:RE
TURN
```

```
2430 REM JUMP ROUTINE!
```

```
2440 "=:PRINTCHR$(131)"Not likely! I might break my neck!":RETURN
```

```
2450 REM WAIT ROUTINE
```

```
2460 K=1
```

```
2464 PRINTCHR$(129);"N.K.":WAIT30
```

```
2470 IFPZ=65THENPZ=66:
```

```
2474 IFPZ=66THENPRINT"The footman opens the door & I get out"
```

```
2475 IFPZ=66THENPRINT"The coach disappears by magic!":PZ=36:
```

```
2477 IFPZ=67THENPZ=68:RETURN
```

```
2480 RETURN
```

```
2490 REM DANCE ROUTINE
```

```
2500 IFPZ=55THENPZ=72:PRINT"The prince likes that!":K=1:RETURN
```

```
2505 PRINTCHR$(129)"I'm not dancing here!":K=1:
```

```
2510 REM ROUTINE TO CHECK FOR MIDNIGHT
```

```
2520 JZ=JZ+1
```

```
2530 PRINTCHR$(129)"BEWARE It is close to midnight!"
```

```
2540 IFJZ>STHENX$(1)="My clothes turn to rags and I get   arreste
d"
```

```
2541 IFJZ>STHEN GOSUB1890
```

```
2550 PRINT 6-JZ;"moves till midnight!"
```

```
2555 RETURN
```

```
2560 REM SUBROUTINE FOR RUN!
```

```
2570 K=1
```

```
2571 IFPZ<>72THENPRINTCHR$(132)"I'm too tired to run at the moment!
":RETURN
```

```
2580 PRINTCHR$(129);"I run away from the ball as fast as I"
```

```
2583 PRINTCHR$(129)"can. I must have left my slipper"
```

```
2585 PRINT" behind. It strikes 12pm. and my   clothds turn to r
ags"
```

```
2586 MZ=19:PZ=63:RETURN
```

```
2590 REM ROUTINE FOR VISIT
```

```
2600 K=1:PRINTCHR$(129)"Two of the prince's servants knock on"
```

```
2602 PRINTCHR$(129)"the door and s'y that the prince will"
```

```
2604 PRINTCHR$(129)"MARRY the girl who can wear the Glass"
```

```
2606 PRINTCHR$(129)"slipper!!!"
```

```
2610 RETURN
```

```
2620 REM SUBROUTINE FOR WEARING SHOE
```

```
2630 K=1
```

```
2640 IFMZ=19THENPRINTCHR$(131)"THE SLIPPER FITS!"
```

```
2642 IFMZ=19THENPRINTCHR$(129)"My fairy godmother reappears & tells
"
```

```
2644 IFMZ=19THENPRINTCHR$(129)"me to get the wedding dress!":MZ=20:
```

```
2650 IFMZ=21THENPRINTCHR$(129)"My fairy godmother reappears and"
```

```
2652 IFMZ=21THENPING:PRINTCHR$(135)"waves her MAGIC WAND!":PZ=67:
```

```
2660 PRINT"Not yet !!!!!!":
```

```
2670 REM GET DRESS ROUTINE
```

```
2680 K=1
```

```
2690 IFMZ<>20THENX$(1)="My UGLY sister comes in and hits me":GOSUB1
890
```

```
2700 MZ=21:K=1:RETURN
```

```
3000 CLS:PAPER0:PLOT0,1,17
```

```
3010 PLOT0,25,17
```

```
3020 FORX=1TO2
```

```
3030 PLOT1,X+6,14:PLOT2,X+6,X:PLOT14,X+6,"CINDERELLA":NEXT
```

```
3040 PLOT1,5,3:PLOT10,5,"An adventure game"
```

```
3050 PLOT1,12,5:PLOT8,12,"<C> 1983 S.W. Lucas"
```

```
3060 PLOT1,22,2:PLOT5,22,"Press <SPACE BAR> to continue."
```

```
3070 REPEAT
```

```
3080 BETA$
```

```
3090 UNTILAS=" ":CLS
```

```
3100 PLOT0,1,19:PLOT0,25,19
```

```
3110 FORX=1TO2:PLOT1,X+6,14:PLOT2,X+6,X:PLOT14,X+6,"CINDERELLA":NEX
T
```

```
3111 PLOT1,10,1:PLOT2,10,"My name is Cinderella":PLOT1,12,6
```

```
3112 PLOT2,12,"Since my mother died, I have lived":PLOT1,13,6
```

```
3113 PLOT2,13,"with my elderly father and my two":PLOT1,14,6
```

```
3114 PLOT2,14,"sisters in a large house two miles":PLOT1,15,6
```

```
3115 PLOT2,15,"from the Royal Palace. My sisters are":PLOT1,16,6
```

```
3116 PLOT2,16,"very CRUEL and will not let me go to":PLOT1,17,6
```

```
3117 PLOT2,17,"the Royal Ball which is to be held in":PLOT1,18,6
```

```
3118 PLOT2,18,"the Royal Palace shortly.":PLOT1,22,12:PLOT2,22,5
```

```
3119 PLOT5,22,"Please help me !!!!!"
```

```
3120 FORT=1TO34:READX
```

```
3130 MUSIC1,3,X,0
```

```
3131 PLAY1,0,4,500
```

```
3135 WAIT10
```

```
3140 NEXT
```

```
3150 PLAY0,0,0,0
```

```
3160 RETURN
```

The Entertainer Listing

By J. P. Hibbins

```

1 PRINTCHR$(17)
10 REM THE ENTERTAINER RAG
20 REM BY SCOTT JOPLIN
30 REM
40 REM COPYRIGHT 1983 J. P. HIBBINS
50 REM
60 S=18:O=S/2:CR=S/4
65 L=3
70 C=1:CC=2:D=3:DD=4:E=5:F=6:FF=7:G=8:GG=9:A=10:AA=11:B=12
75 REM
80 REM TITLE PAGE
90 REM
100 CLS:PAPER0:INK5
110 FORX=1TO9:PRINT:NEXT
120 PRINTCHR$(27);"J";CHR$(27);"D";SPC(5);"SCOTT JOPLIN:"
130 PRINTCHR$(27);"J";CHR$(27);"F";SPC(5);"SCOTT JOPLIN:"
135 PRINT:PRINT
140 PRINTCHR$(27);"J";CHR$(27);"C";SPC(7);"THE ENTERTAINER RAG"
150 PRINTCHR$(27);"J";CHR$(27);"B";SPC(7);"THE ENTERTAINER RAG"
160 PRINT:PRINT:PRINT
170 REM
180 REM ::::: INTRODUCTION :::::
190 REM
200 MUSIC1,5,D,L:MUSIC2,4,D,L:PLAY3,0,X,Y:WAITS
210 MUSIC1,5,E,L:MUSIC2,4,E,L:WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS
220 MUSIC1,4,A,L:MUSIC2,3,A,L:WAITQ:MUSIC1,4,B,L:MUSIC2,3,B,L:WAITS
230 MUSIC1,4,G,L:MUSIC2,3,G,L:WAITQ
300 MUSIC1,4,D,L:MUSIC2,3,D,L:WAITS
310 MUSIC1,4,E,L:MUSIC2,3,E,L:WAITS:MUSIC1,4,C,L:MUSIC2,3,C,L:WAITS
320 MUSIC1,3,A,L:MUSIC2,2,A,L:WAITQ:MUSIC1,3,B,L:MUSIC2,2,B,L:WAITS
330 MUSIC1,3,G,L:MUSIC2,2,G,L:WAITQ
350 MUSIC1,3,D,L:MUSIC2,2,D,L:WAITS:MUSIC1,3,E,L:MUSIC2,2,E,L:WAITS
360 MUSIC1,3,C,L:MUSIC2,2,C,L:WAITS:MUSIC1,2,A,L:MUSIC2,1,A,L:WAITQ
370 MUSIC1,2,B,L:MUSIC2,1,B,L:WAITS:MUSIC1,2,A,L:MUSIC2,1,A,L:WAITS
380 MUSIC1,2,GG,L:MUSIC2,1,GG,L:WAITS:MUSIC1,2,G,L:MUSIC2,1,G,L:WAITQ
385 PLAYO,0,0,0:L=L+3
390 WAITQ:MUSIC1,4,G,L:MUSIC2,3,B,L:MUSICS,0,G,L
400 PLAY7,0,X,Y:WAITQ
405 L=L-5
410 MUSIC1,3,D,L:MUSIC2,2,B,L:MUSICS,2,G,L:WAITS:MUSIC1,3,DD,L:WAITS
420 REM
430 REM :: FIRST DBL BAR ::
440 REM
445 IFH=1 THENH=1
450 MUSIC1,3,E,L:MUSIC2,2,C,L:MUSICS,2,C,L:WAITS:MUSIC1,4,C,L:WAITS
460 MUSIC2,3,C,L:MUSICS,2,E,L:WAITS:MUSIC1,3,E,L:WAITS
470 MUSIC1,4,C,C:MUSIC2,2,G,L:MUSICS,1,G,L:WAITQ
480 MUSIC1,3,E,L:MUSIC2,3,C,L:MUSICS,2,AA,L:WAITS:MUSIC1,4,C,L:WAITS
490 REM
500 MUSIC2,2,F,L:MUSICS,1,F,L:WAITQ:MUSIC2,3,C,L:MUSICS,2,A,L:WAITQ
510 MUSIC2,2,E,L:MUSICS,1,E,L:WAITS
513 IFR=1THENRETURN
516 MUSIC1,5,C,L:MUSIC2,4,E,L:WAITS:L=L+1
520 MUSIC1,5,D,L:MUSIC2,4,F,L:MUSICS,2,G,L:WAITS:L=L+1
530 MUSIC1,5,DD,L:MUSIC2,4,FF,L:WAITS:L=L+1
540 MUSIC1,5,E,L:MUSIC2,4,G,L:MUSICS,1,G,L:WAITS

```

```

550 MUSIC1,5,C,L:MUSIC2,4,E,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,F,L:MUSICS,2,G,L
560 WAITS:MUSIC1,5,E,L:MUSIC2,4,G,L:WAITS:MUSICS,1,G,L:WAITS
570 MUSIC1,4,B,L:MUSIC2,4,D,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,F,L:MUSICS,2,B,L
580 WAITQ:MUSIC1,5,C,L:MUSIC2,4,E,L:MUSICS,2,C,L:WAITQ
585 IFR=2THENRETURN
590 MUSICS,1,G,L:WAITQ:MUSICS,2,C,L:WAITQ:L=L-2
600 MUSIC1,3,D,L:MUSIC2,2,B,L:MUSICS,2,G,L:WAITS:MUSIC1,3,DD,L:WAITS
610 R=1:L=L-1:GOSUB430
615 WAITS
620 L=L+1:MUSIC1,4,A,L:MUSIC2,4,C,L:MUSICS,2,DD,L:WAITS:L=L+1
630 MUSIC1,4,G,L:MUSIC2,3,G,L:WAITS:L=L+1
640 MUSIC1,4,FF,L:MUSIC2,3,FF,L:MUSICS,1,D,L:WAITS
650 MUSIC1,4,A,L:MUSIC2,3,A,L:WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L:MUSICS,2,FF,L
655 WAITS
660 MUSIC1,5,E,L:MUSIC2,4,E,L:WAITS:MUSICS,2,D,L:WAITS
670 MUSIC1,5,D,L:MUSIC2,4,D,L:WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L:MUSICS,2,A,L
675 WAITS:MUSIC1,4,A,L:MUSIC2,3,A,L:WAITS
680 MUSIC1,5,D,L:MUSIC2,4,F,L:MUSICS,2,B,L:WAITQ
690 MUSICS,1,G,L:WAITQ:MUSICS,1,A,L:WAITQ:L=L-1:MUSIC1,3,D,L:MUSIC2,2,B,L
700 MUSICS,1,B,L:WAITS:L=L-1:MUSIC1,3,DD,L:WAITS
710 R=2:GOSUB430
720 MUSICS,2,G,L:WAITQ:MUSICS,3,C,L:WAITQ:PLAY3,0,X,Y
730 MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,D,L:WAITS
740 MUSICS,2,C,L:PLAY7,0,X,Y:MUSIC1,5,E,L:MUSIC2,4,E,L:WAITS
750 MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,D,L:MUSICS,3,C,L
760 WAITS:MUSIC1,5,E,L:MUSIC2,4,E,L:WAITS:MUSICS,1,AA,L:WAITS
770 MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,D,L:MUSICS,3,E,L
780 WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS
790 MUSIC1,5,E,L:MUSIC2,4,E,L:MUSICS,1,A,L:WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L
800 WAITS:MUSIC1,5,D,L:MUSIC2,4,D,L:MUSICS,3,F,L:WAITS:MUSIC1,5,E,L
810 MUSIC2,4,E,L:WAITS:MUSICS,1,GG,L:WAITS:MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS
820 MUSIC1,5,D,L:MUSIC2,4,D,L:MUSICS,3,F,L:WAITS
830 MUSIC1,5,C,L:MUSIC2,4,C,L:WAITS:MUSIC1,5,E,L:MUSIC2,4,G,L:MUSICS,1,G,L
840 WAITS:MUSIC1,5,C,L:MUSIC2,4,E,L:WAITS:MUSIC1,5,D,L:MUSIC2,4,F,L
850 MUSICS,3,C,L:WAITS:MUSIC1,5,E,L:MUSIC2,4,G,L:WAITS
860 MUSICS,1,G,L:WAITS:MUSIC1,4,B,L:MUSIC2,4,D,L:WAITS
870 MUSIC1,5,D,L:MUSIC2,4,F,L:MUSICS,2,B,L:WAITQ
880 IFR=1THENR=0
890 MUSIC1,5,C,L:MUSIC2,4,E,L:MUSICS,2,C,L:WAITQ:MUSICS,1,G,L:WAITQ
895 L=L-1:MUSIC1,1,A,L:WAITQ
900 L=L-1:MUSIC1,3,D,L:MUSIC2,2,B,L:MUSICS,1,B,L:WAITS
910 MUSIC1,3,DD,L:WAITS:R=0:Z=1:L=L-2:GOTO430
940 REM
950 REM 2ND DBL BAR
960 REM
965 R=0
967 IFH=1THENGOTO1550
970 MUSIC1,5,C,L:MUSIC2,4,E,L:MUSICS,2,C,L:WAITQ:MUSICS,1,G,L+1:WAITQ
980 MUSICS,1,C,L+1:WAITS
990 MUSIC1,4,E,L:MUSIC2,4,C,L:WAITS:PLAY3,0,X,Y
992 MUSIC1,4,F,L:MUSIC2,4,D,L:WAITS:MUSIC1,4,FF,L:MUSIC2,4,DD,L:WAITS
995 PLAY7,0,X,Y
1000 REM BACK TO HERE LATER!
1010 MUSIC1,4,G,L:MUSIC2,4,E,L:MUSICS,1,C,L:WAITQ
1020 MUSIC1,4,A,L:MUSIC2,4,E,L:MUSICS,2,C,L:WAITS:MUSIC1,4,G,L:MUSIC2,4,E,L
1022 IFK=1THEN1065

```


The Entertainer Listing

```

1023 IFY=1 ANDR=1THEN1026
1025 IFR=1THENRETURN
1026 WAITS
1029 REM
1030 MUSIC3,1,G,L:WAITS:MUSIC1,4,E,L:MUSIC2,4,C,L:WAITS
1040 MUSIC1,4,F,L:MUSIC2,4,D,L:MUSIC3,3,C,L:WAITS:MUSIC1,4,FF,L:MUSIC2,4,DD,L
1050 WAITS
1055 IFR=1 ANDY=1THENK=1:GOTO1000
1060 R=1:GOSUB1000
1065 WAITS
1067 IFX=1THENRETURN
1070 MUSIC3,1,G,L:WAITS:MUSIC1,4,E,L:WAITS
1080 MUSIC1,4,C,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS:MUSIC1,3,G,L:WAITS
1090 MUSIC1,3,A,L:MUSIC2,1,F,L:MUSIC3,1,F,L:WAITS:MUSIC1,3,B,L:WAITS
1100 MUSIC1,4,C,L:MUSIC2,3,F,L:MUSIC3,2,A,L:WAITS:MUSIC1,4,D,L:WAITS
1111 MUSIC1,4,E,L:MUSIC2,2,F,L:MUSIC3,2,F,L:WAITS:MUSIC1,4,D,L:WAITS
1120 MUSIC1,4,C,L:MUSIC2,3,F,L:MUSIC3,2,GG,L:WAITS:MUSIC1,4,D,L:WAITS
1125 IFY=1THENRETURN
1130 MUSIC1,3,G,L:MUSIC2,2,E,L:MUSIC3,2,E,L:WAITS:MUSIC1,4,E,L:WAITS
1140 MUSIC1,4,F,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,G,L:WAITS
1150 MUSIC1,4,A,L:MUSIC2,1,G,L:MUSIC3,1,G,L:WAITS:MUSIC1,4,G,L:WAITS
1160 MUSIC1,4,E,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,F,L:WAITS
1165 R=0
1170 X=1:GOSUB1000
1180 X=0:MUSIC3,2,E,L:WAITS:MUSIC1,4,G,L:WAITS
1190 MUSIC1,4,A,L:MUSIC2,2,DD,L:MUSIC3,2,DD,L:WAITS:MUSIC1,4,AA,L:WAITS
1200 MUSIC1,4,B,L:MUSIC2,4,D,L:MUSIC3,2,D,L:WAITS:PLAY6,0,0,0:WAIT.2
1210 PLAY7,0,0,0:WAITS:MUSIC3,2,G,L:WAITS:PLAY6,0,0,0:WAIT.2:PLAY7,0,0,0
1220 MUSIC2,4,C,L:WAITS:MUSIC3,2,D,L:WAITS:MUSIC1,4,A,L:WAITS
1230 MUSIC1,4,FF,L:MUSIC2,3,C,L:MUSIC3,2,A,L:WAITS:MUSIC1,4,D,L:WAITS
1240 MUSIC1,4,G,L:MUSIC2,3,B,L:MUSIC3,2,G,L:WAITO
1250 MUSIC3,1,F,L:WAITO:MUSIC3,1,E,L:WAITS:MUSIC1,4,E,L:MUSIC2,4,C,L:WAITS
1260 MUSIC1,4,F,L:MUSIC2,4,D,L:MUSIC3,1,D,L:WAITS:MUSIC1,4,FF,L:MUSIC2,4,DD,L
1270 WAITS:X=0:Y=1:GOSUB1000
1280 MUSIC1,4,C,L:MUSIC2,2,E,L:MUSIC3,2,E,L:WAITO:MUSIC2,3,E,L:MUSIC3,2,G,L
1290 WAITO:MUSIC3,2,C,L:MUSIC2,2,C,L:WAITS:MUSIC1,3,G,L:WAITS
1300 L=L-1:MUSIC1,3,FF,L:MUSIC2,3,E,L:MUSIC3,2,AA,L:WAITS:MUSIC1,3,G,L:WAITS
1310 L=L-1:MUSIC1,4,C,L:MUSIC2,3,A,L:MUSIC3,2,F,L:WAITO:PLAY3,0,0,0:WAIT.2
1320 PLAY7,0,0,0:Y:MUSIC1,3,A,L:MUSIC2,3,F,L:WAITS:MUSIC1,4,C,L:WAITS
1330 MUSIC2,3,DD,L:MUSIC3,2,FF,L:WAITS:MUSIC1,3,A,L:WAITS:PLAY1,0,0,0:WAIT.2
1340 PLAY7,0,0,0:X,Y:MUSIC1,4,C,L:WAITS:MUSIC1,3,A,L:WAITS
1350 MUSIC1,3,G,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,C,L:WAITS
1360 PLAY1,0,0,0:X,Y:WAIT.2:PLAY7,0,0,0,Y:MUSIC1,4,E,L:WAITS:MUSIC1,4,G,L:WAITS
1370 PLAY1,0,0,0:X,Y:WAIT.2:PLAY7,0,0,0,X,Y:WAITS:MUSIC1,4,E,L:WAITS
1380 PLAY1,0,0,0:X,Y:WAIT.2:PLAY7,0,0,0,Y:MUSIC1,4,C,L:WAITS:MUSIC1,3,G,L:WAITS
1390 MUSIC1,3,A,L:MUSIC2,3,FF,L:MUSIC3,2,D,L:WAITO:PLAY1,0,0,0:WAIT.2
1400 PLAY7,0,0,0,X,Y:MUSIC1,4,C,L:WAITO:MUSIC1,4,E,L:MUSIC2,3,F,L:MUSIC3,1,G,L
1410 WAITS:MUSIC1,4,D,L:WAITS:MUSIC3,1,B,L:WAITS:MUSIC1,4,C,L:MUSIC2,3,E,L
1420 WAITS:MUSIC3,2,C,L:WAITO:MUSIC3,1,G,L:WAITO
1430 IFR=1THEN1500
1440 MUSIC3,1,E,L:WAITS:MUSIC1,4,E,L:MUSIC2,4,C,L:WAITS
1450 MUSIC1,4,D,L:MUSIC2,4,D,L:MUSIC3,1,D,L:WAITS:MUSIC1,4,FF,L
1460 MUSIC2,4,DD,L:WAITS
1470 P=1:K=0:R=0:Y=0:X=0:L=L+2:GOSUB1000
1500 MUSIC3,1,C,L:WAITO
1505 X=0:K=0:Y=0:R=0
1507 PLAY1,0,0,0,X,Y:MUSIC1,3,D,L:WAITS:MUSIC1,3,DD,L:WAITS:PLAY7,0,0,0,X,Y

```

```

1510 IFH=0THENL=L-1:GOTO430
1550 MUSIC1,5,C,L:MUSIC2,4,E,L:MUSIC3,2,C,L:WAITO:MUSIC3,1,G,L:WAITO
1555 PLAY4,0,0,0,X,Y:WAIT.2:PLAY7,0,0,0,X,Y:MUSIC3,1,C,L:WAITO
1560 PLAY0,0,0,0:WAITO:PLAY7,0,0,0,X,Y
1600 REM REPEAT TO HERE
1601 IFU=1THENW=1
1610 MUSIC1,3,F,L:MUSIC2,3,D,L:MUSIC3,1,F,L:WAITO:MUSIC1,3,E,L:MUSIC2,3,CC,L
1620 MUSIC3,2,A,L:WAITS:MUSIC1,3,F,L:MUSIC2,3,D,L:WAITS
1630 MUSIC3,1,A,L:WAITS:MUSIC1,3,E,L:MUSIC2,3,CC,L:WAITS
1640 MUSIC1,3,F,L:MUSIC2,3,D,L:MUSIC3,2,A,L:WAITO
1645 MUSIC3,1,F,L:PLAY4,0,0,0,X,Y:WAITS
1650 MUSIC1,3,A,L:MUSIC2,3,F,L:PLAY5,0,0,0,X,Y:WAITS
1660 PLAY7,0,0,0,X,Y:MUSIC1,4,D,L:MUSIC3,2,A,L:WAITS:MUSIC1,3,A,L:WAITS
1670 MUSIC1,4,C,L:MUSIC2,1,A,L:MUSIC3,1,A,L:WAITS:MUSIC1,4,D,L:WAITS
1680 MUSIC1,4,C,L:MUSIC2,2,A,L:MUSIC3,2,F,L:WAITS:MUSIC1,3,A,L:WAITS
1690 MUSIC1,3,G,L:MUSIC2,3,E,L:MUSIC3,2,C,L:WAITO
1700 MUSIC1,3,FF,L:MUSIC2,3,DD,L:MUSIC3,3,C,L:WAITS:MUSIC1,3,G,L
1710 MUSIC2,3,E,L:WAITS:MUSIC3,1,G,L:WAITS:MUSIC1,3,FF,L:MUSIC2,3,DD,L:WAITS
1720 MUSIC1,3,G,L:MUSIC2,3,E,L:MUSIC3,3,C,L:WAITO
1730 MUSIC1,2,C,L:MUSIC2,2,C,L:WAITS:MUSIC1,4,C,L:WAITS
1740 MUSIC1,4,G,L:MUSIC2,1,G,L:MUSIC3,3,C,L:WAITS:MUSIC1,4,C,L:WAITS
1750 MUSIC1,4,D,L:MUSIC2,1,G,L:MUSIC3,1,G,L:WAITS:MUSIC1,4,E,L:WAITS
1760 MUSIC1,4,D,L:MUSIC2,3,C,L:MUSIC3,2,E,L:WAITS:MUSIC1,4,C,L:WAITS
1765 IFW=1THENGOTO1925
1770 MUSIC1,4,D,L:MUSIC2,3,B,L:MUSIC3,1,G,L:WAITO
1780 MUSIC1,4,CC,L:MUSIC2,3,AA,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,D,L
1790 MUSIC2,3,B,L:WAITS:MUSIC3,1,B,L:WAITS:MUSIC1,4,CC,L:MUSIC2,3,AA,L:WAITS
1800 MUSIC1,4,D,L:MUSIC2,3,B,L:MUSIC3,2,G,L:WAITO
1810 MUSIC1,1,G,L:MUSIC2,1,G,L:MUSIC3,1,G,L:WAITS:MUSIC1,4,F,L:WAITS
1820 MUSIC1,4,A,L:MUSIC2,3,B,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,F,L:WAITS
1830 MUSIC1,4,G,L:MUSIC2,2,D,L:MUSIC3,2,D,L:WAITS:MUSIC1,4,A,L:WAITS
1840 MUSIC1,4,G,L:MUSIC2,3,B,L:MUSIC3,2,G,L:WAITS:MUSIC1,4,F,L:WAITS
1850 L=L+2:MUSIC1,5,C,L:MUSIC2,3,FF,L:MUSIC3,2,DD,L:WAITS
1860 PLAY6,0,0,0,X,Y:WAIT.2:PLAY7,0,0,0,X,Y:WAITS
1870 PLAY0,0,0,0:WAIT.2:PLAY7,0,0,0,X,Y:WAITS*3:PLAY0,0,0,0:WAITS
1880 MUSIC1,4,A,L:PLAY7,0,0,0,X,Y:WAITO:MUSIC1,4,G,L:MUSIC2,3,C,L:MUSIC3,2,E,L
1890 WAITO:L=L-3:PLAY1,0,0,0,X,Y:MUSIC1,3,G,L:WAITS:MUSIC1,3,E,L:WAITS
1895 MUSIC1,3,G,L:WAITO:MUSIC1,3,E,L:WAITO:MUSIC2,3,D,L:MUSIC3,1,F,L
1900 IFU=0THENPLAY7,0,0,0,X,Y:U=1:GOTO1600
1925 MUSIC1,3,A,L:MUSIC2,2,F,L:MUSIC3,1,F,L:WAITS:MUSIC1,3,GG,L:WAITS
1930 MUSIC1,3,A,L:MUSIC2,2,D,L:MUSIC3,1,D,L:WAITS:MUSIC1,4,G,L:MUSIC2,3,A,L
1940 WAITS:MUSIC3,1,E,L:WAITS:MUSIC1,4,F,L:WAITS:MUSIC3,1,F,L:WAITS
1950 MUSIC1,4,C,L:WAITS:MUSIC1,4,E,L:MUSIC2,3,G,L:MUSIC3,1,G,L:WAITS
1959 MUSIC1,4,DD,L:WAITS:MUSIC1,4,E,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS
1960 MUSIC1,4,A,L:WAITS:MUSIC2,3,DD,L:MUSIC3,2,FF,L
1965 WAITS:MUSIC1,5,C,L:WAITS:MUSIC1,4,G,L:MUSIC2,3,E,L:MUSIC3,2,G,L:WAITS
1968 MUSIC1,4,E,L:WAITS:MUSIC1,4,C,L:MUSIC2,3,FF,L:MUSIC3,1,A,L:WAITO
1970 PLAY4,0,0,0,X,Y:WAIT.2:MUSIC3,1,D,L:PLAY7,0,0,0,X,Y:WAITO
1975 MUSIC1,4,E,L:MUSIC2,3,F,L:MUSIC3,1,G,L:WAITS:MUSIC1,4,D,L:WAITS
1980 MUSIC3,1,B,L:WAITS:MUSIC1,4,C,L:MUSIC2,3,E,L:WAITS:MUSIC3,2,C,L:WAITO
1983 IFY=0THEN1986 ELSE1993
1986 PLAY1,0,0,0,X,Y:MUSIC1,3,G,L:WAITS:MUSIC1,3,E,L:WAITS:MUSIC1,3,G,L:WAITO
1990 MUSIC1,3,E,L:WAITO:TY=1:U=0:W=0:PLAY7,0,0,0,X,Y:GOTO1600
1993 L=L+1
1995 MUSIC3,1,G,L:WAITO:L=L+1:MUSIC1,5,C,L:MUSIC2,4,E,L:MUSIC3,1,C,L:WAITO
2000 PLAY0,0,0,0
2001 PRINTCHR$(17)

```

Grand Prix Listing

By M. Caldwell

```

1 CLS
5 GOSUB60000:INTRO
10 GOSUB10000
20 Y=12
30 F=37
70 PLOT0,0,17:PLOT5,0,19:PLOT10,0,18
90 PLOT0,7,"=====
100 PLOT0,17,"=====
110 PLOT12,3,"MILES "
120 PLOT8,20,"HIGHEST MILEAGE "+STR$(H)
130 PLOT5,22,"DRIVEN BY "+X$
1200 PLOT0,6,A$:PLOT0,18,A$
1300 L$=LEFT$(A$,1):R$=RIGHT$(A$,37)
1400 A$=R$+L$
1500 PLOT18,Y," "
1510 PLOT0,8,B$:PLOT0,16,B$
1520 PLOT0,9,C$:PLOT0,15,D$
1530 PLOT0,10,D$:PLOT0,13,G$:PLOT0,14,C$
1550 PLOT0,11,E$:PLOT0,12,F$
1600 Y$=KEY$
1620 L2$=LEFT$(C$,1):R2$=RIGHT$(C$,37):L5$=LEFT$(F$,1):R5$=RIGHT$(F$,37)
1630 L3$=LEFT$(D$,1):R3$=RIGHT$(D$,37):L6$=LEFT$(G$,1):R6$=RIGHT$(G$,37)
1640 L1$=LEFT$(B$,1):R1$=RIGHT$(B$,37):L4$=LEFT$(E$,1):R4$=RIGHT$(E$,37)
1650 C$=R2$+L2$:E$=R4$+L4$
1660 D$=R3$+L3$:F$=R5$+L5$
1670 B$=R1$+L1$:G$=R6$+L6$
1700 IFY$=""THEN1900
1800 M=ASC(Y$)
1900 IFM=11THENY=Y-1
2000 IFM=10THENY=Y+1
2100 IFSCRN(18,Y)<>32THENEXPLODE:S=S-10
2200 PLOT18,Y,">"
2210 F=F-.065
2220 PLOT6,0,16
2300 IFF<1THENZAP:ZAP:ZAP:ZAP:ZAP:GOTO40000
2400 PLOT19,3,STR$(S)
2500 IFS<400THENS=S+1ELSESS=S+2
2600 IFS=400THEN20000
2700 IFS>400THENF=F-.080
9000 GOTO1200
10000 A$=": : : : : : : : "
10100 B$="K K K K K K K K "
10200 C$=" K K K K K K K K "
10300 D$=" K K K K K K K K "
10400 E$=" K K K K K K K K "
10500 F$=" K K K K K K K K "
10600 G$=" K K K K K K K K "
11010 RETURN
20000 CLS:ZAP:F=37
20010 PRINT:PRINT:PRINT"IT GETS HARDER NOW !!!!!"
20020 CLS:B$="K K K K K K K K "
20030 C$=" K K K K K K K K "
20040 D$=" K K K K K K K K "
20050 F$=" K K K K K K K K "
20060 G$=" K K K K K K K K "
20070 E$=" K K K K K K K K "
20080 Y=12:GOTO70
40000 CLS
40100 PRINT:PRINT:PRINT"NO FUEL LEFT,BUT YOU GOT A MILEAGE OF"
40200 PRINT:PRINTS:" MILES"
40300 IFS>HTHENPRINT:PRINT"THAT IS BETTER THAN SOMEBODY CALLED "
40400 IFS<HTHENPRINT:PRINT:PRINT"THAT MILEAGE DOES NOT BEAT THAT OF "
40500 PRINT:PRINTX$
40600 IFS>HTHENH=S:INPUT"INPUT YOUR NAME,CHAMP!!!!!":X$
40700 PRINT:PRINT:PRINT"LIKE ANOTHER GO Y\N?":GETA$
40800 IFA$="Y"ORAS$="Y"THENS=0:CLS:GOTO10
40900 IFA$="N"ORAS$="n"THENEND
41000 GOTO40700
60000 CLS:IFS/600THENH=600:X$="MICHAEL CALDWELL"
60010 PRINT:PRINT:PRINT:PRINTSPC(10)"GRAND PRIX"
60020 PLOT14,5,CHR$(96)+" MICHAEL CALDWELL 1893"
60030 PRINT:PRINT:PRINT:PRINT"YOU ARE THE 'X' AND MUST DODGE THE "
60040 PRINT:PRINT"'K' USING YOUR CURSER UP AND DOWN "
60050 PRINT:PRINT"KEYS.THERE IS NO LIVE RESTRICTION "
60060 PRINT:PRINT"BUT THERE IS ONLY A LIMITED FUEL "
60070 PRINT:PRINT"SUPPLY"
60080 PRINT:PRINT:PRINTSPC(12)"Good Luck"
60090 PRINT:PRINT:PRINT:PRINT"PRESS ANY KEY TO START THE GAME!!!!"
60100 FORA=46576TO46585:READZ:POKEA,Z:NEXT
60110 FORA=46680TO46688:READZ:POKEA,Z:NEXT
60130 DATA6,32,38,63,63,38,32,6,0
60140 DATA0,6,0,38,63,63,38,0,6,0
60150 PRINTCHR$(17)CHR$(6)
60155 GETA$:PAPER0:INK7:CLS:RETURN

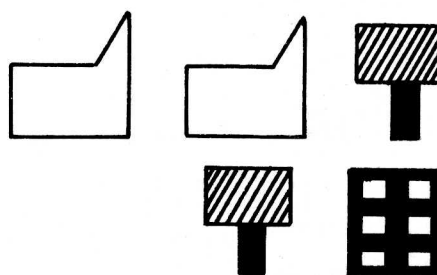
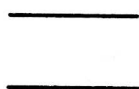
```

Counting Shapes

By S. G. Roper

This program is designed for young Oric owners and allows them to practice shape recognition and counting. Various graphic shapes are displayed on the screen: Horses, trees, people etc. The user then has to count them in various ways. Hours of fun are guaranteed!

**How many
Trees ?**



```

0 REM *****
1 REM ***** COUNTING SHAPES *****
2 REM *** WRITTEN BY S.G.ROPER *****
3 REM ***** COPYRIGHT 1983 *****
4 REM *****
5 CLS:POKE #26A,10:PAPER 0:INK 3:PRINT CHR$(20):POKE 524,255
6 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
7 PRINT CHR$(4);CHR$(27);"N";CHR$(27);"A";
8 PRINT" STOP THE TAPE STOP THE TAPE";CHR$(4);PING:WAIT 30:PING
10 GOSUB 1000
13 REM ** SELECT LEVEL OF PROBLEM **
14 CLS:PRINT CHR$(130);"*****";PRINT
15 PRINT:PRINTCHR$(133)" " * 30;CHR$(133) COUNTING SHAPES * * * * *;PRINT
16 PRINTCHR$(131)" ( )";CHR$(129)"BY S.G.ROPER";CHR$(131)" ( ) "
17 PRINT:PRINTCHR$(134)" "
18 PRINT:PRINTCHR$(135)" "
19 PRINT:PRINTCHR$(132)" "
20 PRINT:PRINTCHR$(134)" "
21 PRINT:PRINTCHR$(130)" "
22 PRINT:IF Z$="C" THEN GOSUB 5000;GOTO 25
23 PRINT"WOULD YOU LIKE TO ADD LARGE NUMBERS";PRINT
24 PRINT" SMALL NUMBERS OR INDIVIDUAL ITEMS";PRINT
25 GOSUB 4000
26 PRINT CHR$(129)" "
27 PRINTCHR$(129)"PRESS 'I' for individual items"
28 PRINT CHR$(129)" "
29 PRINT CHR$(129)" "
30 GET T$;IF T$="L" OR T$="S" OR T$="I" THEN 33
31 IF T$="Z" THEN L=0:M=0:GOTO 14
32 GOTO 30
33 CLS:PAPER 0:INK 3:PRINT:GOSUB 1300
34 REM ** RAND NUMBER & LINES GEN **
35 WAIT RND(1)*10
36 IF T$="L" THEN U=8:V=2:M=40:GOTO 46
37 IF T$="S" THEN U=6:V=1:M=10:GOTO 46
40 IF T$="I" THEN U=8:V=2:M=40:GOTO 46
46 PRINT CHR$(4)
48 A=INT(RND(1)*4)+1+V:FOR B=1 TO A
50 C=INT(RND(1)*U)+V
60 REM ** SPACING AND PRINT NUMBER **
64 PRINT CHR$(27)"J";
65 PRINT" ";
70 IF C<10 THEN PRINT" ";
80 IF A=8 THEN PRINT C" ";
85 FOR D=1 TO C
100 REM ** PRINT CHARACTERS **
101 J=INT(RND(1)*7)+1
102 IF J=1 THEN JA=JA+1:PRINT CHR$(27)"B#";
103 IF J=2 THEN JB=JB+1:PRINT CHR$(27)"D#";
104 IF J=3 THEN JC=JC+1:PRINT CHR$(27)"E";
105 IF J=4 THEN JD=JD+1:PRINT CHR$(27)"G#";
106 IF J=5 THEN JE=JE+1:PRINT CHR$(27)"C1/";
107 IF J=6 THEN JF=JF+1:PRINT CHR$(27)"F#";
108 IF J=7 THEN JG=JG+1:PRINT CHR$(27)"A#";
110 NEXT D
120 PRINT:PRINT
130 E=E+C
135 IF E>M THEN 150
140 NEXT B
150 M=M+1
160 PRINT:PRINT CHR$(4);"-----"
165 REM ** GET ANSWER(1,2 DIGIT) **
170 PRINT CHR$(4);CHR$(27)"J ";
175 GOSUB 3000
180 IF E<10 THEN PRINT" ";
182 GET F$:F=VAL(F$)
184 IF ASC(F$)<48 OR ASC(F$)>57 THEN 182
186 PRINT F;
188 IF E<10 THEN H=VAL(F$):GOTO 200
192 GET G$:G=VAL(G$)
194 IF ASC(G$)<48 OR ASC(G$)>57 THEN 192
196 PRINT G;
198 H=F+G:H=VAL(H$)
200 REM ** COMMENT ON ANSWER **
210 IF E=H THEN PRINT" - - - CORRECT ANSWER - - -";CHR$(4)
212 IF E=H THEN N=100:REPEAT:SOUND1,N,12:WAIT2:N=N-5:UNTILN<10:SOUND1,0,1
213 IF E=H THEN L=L+1:GOTO 300
215 PRINT" - SORRY, THE ANSWER IS "E"- " CHR$(4)

```

Counting Shapes Listing

```

220 N=10:REPEAT:SOUND1,N,12:WAIT1:N=N+3:UNTILN>90:SOUND1,0,1:GOTO 300
300 REM ** PRINT SCORE AND CONT? **
302 PRINT:PRINT
306 PRINTCHR$(129) " :SPACE-BAR to continue"
307 PRINTCHR$(129) " PRESS !"
308 PRINTCHR$(129) " :?C: to check score"
315 GET Z$
320 IF ASC(Z$)=32 THEN 33
340 IF Z$="C"THEN 14
360 GOTO 315
400 POKE #26A,3:CLS:END
1000 REM ** CHARACTER GENERATION **
1010 FOR N=46400 TO 46431:READ V:POKE N,V:NEXT
1020 DATA 1,2,4,9,16,21,17,0,32,16,8,36,2,42,34,0
1030 DATA 1,1,15,11,3,2,6,0,32,32,60,52,48,16,24,0
1040 FOR N=46360 TO 46391:READ V:POKE N,V:NEXT
1050 DATA 1,3,7,15,31,1,3,0,0,32,48,56,60,0,32,0
1060 DATA 0,0,4,3,15,10,0,0,0,4,56,62,42,26,0
1070 FOR N=46360 TO 46375:READ V:POKE N,V:NEXT
1080 DATA 0,0,15,15,15,8,0,8,0,8,60,44,36,32,0
1090 FOR N=46344 TO 46351:READ V:POKE N,V:NEXT
1100 DATA 40,56,56,63,44,60,47,0
1110 FOR N=46456 TO 46463:READ V:POKE N,V:NEXT
1120 DATA 0,0,0,62,50,50,62,0
1130 FOR N=46576 TO 46591:READ V:POKE N,V:NEXT
1140 DATA 63,34,63,34,63,34,62,0,63,17,63,17,31,0
1200 RETURN
1300 REM ** ZERO VALUE **
1310 JA=0:JB=0:JC=0:JD=0:JE=0:JF=0:JG=0:JH=0:JL=0:E=0
2000 REM ** WHAT TO COUNT **
2005 AN=INT(RND(1)*7)+1
2008 IF AN=1 THEN PRINT CHR$(130) "
2009 IF AN=2 THEN PRINT CHR$(132) "
2010 IF AN=3 THEN PRINT CHR$(133) "
2011 IF AN=4 THEN PRINT CHR$(135) "
2012 IF AN=5 THEN PRINT CHR$(131) "
2013 IF AN=6 THEN PRINT CHR$(134) "
2014 IF AN=7 THEN PRINT CHR$(129) "
2020 PRINT " COUNT ALL THE "
2025 IF T$="S" OR T$="L" THEN AN=B:GOTO 2100
2030 IF AN=1 THEN AN$="TREES":PRINT AN$ " $"
2040 IF AN=2 THEN AN$="SPIDERS":PRINT AN$ " %"
2050 IF AN=3 THEN AN$="CHALET BUNGALOWS":PRINT AN$ " {} "
2060 IF AN=4 THEN AN$="MEN":PRINT AN$ " *+*"
2070 IF AN=5 THEN AN$="CHURCHES":PRINT AN$ " !/"
2080 IF AN=6 THEN AN$="BLOCKS OF FLATS":PRINT AN$ ">?"
2090 IF AN=7 THEN AN$="ANIMALS":PRINT AN$ "<="
2100 IF AN=B THEN AN$="SHAPES":PRINT AN$
2200 RETURN
3000 REM ** TOTAL NUMBER (ANS) **
3005 IF AN=1 OR AN=9 THEN E=E
3010 IF AN=1 THEN E=JA
3020 IF AN=2 THEN E=JB
3030 IF AN=3 THEN E=JC
3040 IF AN=4 THEN E=JD
3050 IF AN=5 THEN E=JE
3060 IF AN=6 THEN E=JF
3070 IF AN=7 THEN E=JG
3080 IF AN=8 THEN E=JH
3100 RETURN
4000 REM ** MUSIC **
4010 AN$="58:58:1<85"
4020 BM$="4474474444458"
4025 CM$="3333333333333"
4030 FOR N=1 TO LEN(AM$)
4040 AM=ASC(MID$(AM$,N))-48
4050 BM=ASC(MID$(BM$,N))-48
4055 CM=ASC(MID$(CM$,N))-48
4060 MUSIC 1,CM,AM,0:PLAY 7,0,1,5000
4080 WAIT BM*5:NEXT
4100 RETURN
5000 REM ** SCORE **
5010 PRINT:PRINTCHR$(4) CHR$(27)"J SCORE --- "L" OUT OF "M
5020 PRINTCHR$(4):PRINT
5030 RETURN

```

STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS

NOTICE FROM TANSOFT

PRINTER OWNERS – PLEASE NOTE

By January 1984 Oric Printer Accessories, namely pens and paper will be available from Tansoft and most Oric hardware stockists. Prices were not available at the time of going to Press.

STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS STOP PRESS

A Question of Death

By S. W. Lucas

An adventure game for the 48K
ORIC1

Background

I am in an extremely difficult position!

My parents died many years ago and since then I have not been on holiday until this year. After years of saving I have taken a well earned rest on the sunny island of POGLOVIA which is situated off the mainland of ZOBLOVIA.

I have had nothing but trouble since arriving here. On the first day, torrential rain caused widespread flooding and many areas of the island are still under water. My money ran out two days ago and to cap it all, the Mad Emperor of Zoblovia has invaded the island with 10000 troops and most of the inhabitants have been killed.

I have heard that he is in hiding somewhere on the island and is planning to blow the island up if his demands are not met. He is asking for the famed DIAMOND OF POGLOVIA.

I am at the moment floating on a piece of driftwood somewhere on the island and the only chance of saving the island is to GIVE the diamond to the emperor or to kill him. The task is made more difficult by the fact that there are only 150 moves before the island is blown up.

You must give me commands in the form of TWO WORD sentences such as GET BOOK, EAT FOOD etc.
GOOD LUCK

Notes on typing the program in

When entering the program from the listing, you may find difficulty if you do not use ? as an abbreviation for PRINT in long lines. The printer used displays the hash symbol as £ and when entering the program use <shift 3> to replace £.

The WAIT command in line 10185 is there to give you chance to read the instructions. Change the value to suit yourself.

Words Understood

N, S, E, W, UP, DOWN, IN, OUT, GO, CLIMB, LOOK, SCORE, EAT, RUB, LAMP, HELP, DRINK, SAY, SPEAK, PRAY, READ, PHONE, INVENTORY, ROW, SAIL, PADDLE, GO BUS, GO RIDE, RIDE, WAIT, JUMP, DIVE, GET, GRAB, TAKE, GIVE, DROP, LEAVE, PUT, OPEN, UNLOCK, KISS, EXAMINE, SEARCH, KILL, SHOOT, ATTACK, DESTROY, SWIM, EMPEROR, ZOBLOVIA, PINT, BEER,

BAR, SOAP, SUITCASE, KEY, DINGHY, LOCK, PADLOCK, DRIFTWOOD, PUPPY, ICE, NAIL, TRUNKS, NEST, DIAMOND, TEA, CUP, TABLE, LEVER, GUN, BULLET, GIRL

Hints

Read this section only in desperation.

1. There are 103 locations to visit and only 150 moves to succeed in your quest. Don't WAIT around anywhere.

2. There are two possible solutions.

(a) find the diamond and GIVE DIAMOND to the emperor.

(b) SHOOT EMPEROR – make sure that you have both the gun and the bullets.

3. If you are stuck by the boat try GO BOAT or ROW or SAIL.

4. If you are stuck at the bus terminus try GO BUS or RIDE BUS or GO RIDE.

5. If you are stuck at the ravine try JUMP.

6. If you are stuck at the bottom of the steps in Woolworths try CLIMB.

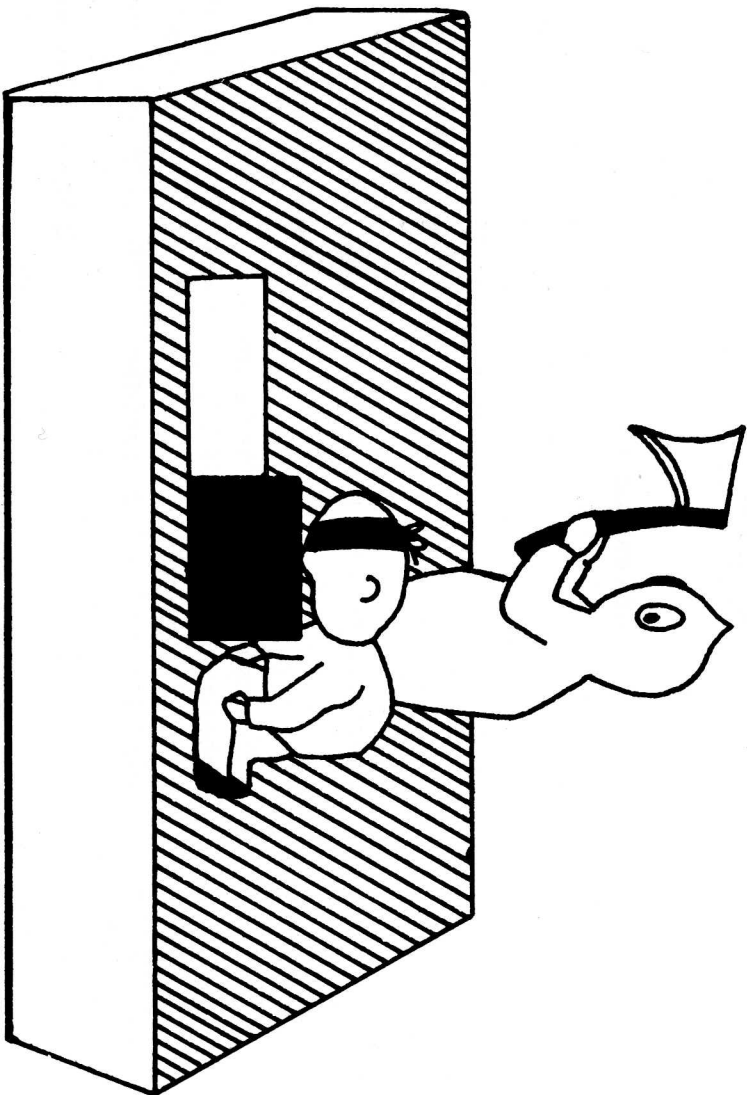
7. If you are stuck at the top try GO DOWN or just DOWN

8. If you can't find the diamond try UNLOCKING the SUITCASE – you will need the key.

9. If you get stuck at the top of a wall try JUMP.

10. By the swimming baths try SWIM, DIVE or JUMP.

A Question of Death Listing



```

10 REM ££ A QUESTION OF DEATH ££
20 REM ££ AN ADVENTURE GAME ££
30 REM £FOR THE 48K ORIC ££
40 REM £ BY S.W. Lucas 1983 £
45 TEXT:GRAB:PAPER0:INK1:CLS
46 PRINT:PRINT:PRINT:PRINT
47 P%=54:G%=0
50 GOSUB10000:PLOT1,25,6:PLOT5,25,"<C> 1983 S.W. Lucas"
70 DIMS%(103,4),X$(35),N$(70),D$(103),G$(60),V$(4),B$(103,1),N$(100)
80 RESTORE:GOSUB1100:GOSUB2000:CLS
90 PRINT
100 PRINTCHR$(131)"I am :-":PRINTD$(P%):A$="":IFS%(P%,1)>0THENA$="North"
105 A=FRE("")
110 IFS%(P%,2)>0ANDLEN(A$)>0THENA$=A$+",South"
120 IFS%(P%,2)>0ANDLEN(A$)=0THENA$="South"
130 IFS%(P%,3)>0ANDLEN(A$)>0THENA$=A$+",East"
140 IFS%(P%,3)>0ANDLEN(A$)=0THENA$="East"
150 IFS%(P%,4)>0ANDLEN(A$)>0THENA$=A$+",West"
160 IFS%(P%,4)>0ANDLEN(A$)=0THENA$="West"
170 IFS%(P%,1)=0ANDS%(P%,3)=0ANDS%(P%,4)=0ANDS%(P%,2)=0THENA$="nowhere ?"
180 IFP%=103ANDAA=1THENCLS:GOTO2600:REM END OF GAME
184 IFG%>149THEN40000
185 IFG%>100THENPRINTCHR$(134):CHR$(17)
190 PRINT:PRINT:PRINTCHR$(130);"I can go :-":PRINTA$:PRINT:PRINT:PRINT
200 PRINT:PRINT"I have ";150-G%:" moves left!"
204 E=0:FORI=1TO20:PP%=0:IFB$(I,1)=P%THENPP%=1
210 IFPP%=1THEN230
220 NEXT:GOTO260
230 IFE=0THENPRINTCHR$(134);"That looks like:-"
240 PRINTB$(I):E=E+1:GOTO220
260 PRINT:PRINTCHR$(133);
261 INPUT"What do I do now ";Z$:B$=LEFT$(Z$,2):C$=LEFT$(Z$,3)
262 CLS:G%=G%+1:PING
263 PRINTCHR$(134);"HANG ON A SEC!"
270 IFC$="LOO"THEN90
280 IFC$="SCO"THENPRINTCHR$(131);"This isn't a game you know!":GOTO90
290 IFC$="EAT"THENPRINTX$(9):GOTO90
291 IFZ$="RUB LAMP"THENPRINTX$(12):GOTO90
292 IFC$="HEL"THENPRINTX$(16):GOTO90
293 IFC$="DRI"THENPRINT"I only drink milk and I can't see any":GOTO90
294 IFC$="SAY"ORC$="SPE"THENPRINTX$(4):GOTO90
295 IFC$="PRA"THENPRINTX$(18):PRINTCHR$(131);"nothing happens":GOTO90
296 IFB$="N"ANDS%(P%,1)<>0THENP%=S%(P%,1):GOTO90
297 IFB$="S"ANDS%(P%,2)<>0THENP%=S%(P%,2):GOTO90
298 IFB$="E"ANDS%(P%,3)<>0THENP%=S%(P%,3):GOTO90
299 IFB$="W"ANDS%(P%,4)<>0THENP%=S%(P%,4):GOTO90
300 IFB$="RE"THENPRINT"I can't see anything to read":GOTO90
301 IFB$="S"ORB$="N"ORB$="E"ORB$="W"THENPRINT"I can't go that way":GOTO90
302 IFC$="PHO"THENPRINT"don't be silly now":GOTO90
303 IFB$="FU"ORB$="PI"THENPRINTX$(22):GOTO90
304 IFC$="BUG"THENPRINTX$(23):GOTO90
305 IFC$="INV"THEN6600
309 IFC$="ROW"ORC$="SAI"ORC$="PAD"ORZ$="GO DINGHY"THEN6900
310 IFB$="UP"ORC$="CLI"ORZ$="GO UP"THEN3000
315 IFC$="DOW"ORZ$="GO DOWN"THEN3100
320 IFC$="OUT"ORZ$="GO OUT"THEN3300
325 IFB$="IN"ORZ$="GO IN"THEN3400
330 IFZ$="GO BUS"ORZ$="GO RIDE"ORC$="RID"THEN3500
335 IFC$="WAI"THENPRINTX$(27):WAIT100:G%=G%+1:GOTO100:REM WAITING UNSES 2 TURNS
340 IFC$="JUM"THEN3550
345 IFC$="DIV"THEN3560
350 IFB$="GE"ORB$="GR"ORB$="TA"THEN4000
360 IFC$="DRO"ORC$="LEA"ORC$="PUT"THEN6000
365 IFC$="OPEN"ANDP%=64THENPRINTX$(26):GOTO100
370 IFC$="UNL"THEN6700
380 IFC$="KIS"THEN6800
385 IFC$="GIV"THEN7000
390 IFC$="EXA"THENPRINT"I can't see anything special":GOTO100
391 IFC$="SEA"THENPRINT"I can't see anything here":GOTO100
395 IFC$="KIL"ORC$="SHO"ORC$="ATT"ORC$="DES"THEN7500
396 IFC$="SWI"THENPRINT"JUST GIVE ME A DIRECTION & I'LL SWIM IF I HAVE TO!"

```

A Question of Death Listing

```

397 IFC$="SWI"THEN100
400 PRINTX$(5):GOTO90
1000 END
1100 FORH=1TO103:READQ$(H):FORD=1TO4:READS$(H,D):NEXTD,H:RETURN
2000 FORD=1TO35:READX$(D):NEXTD
2010 FORH=1TO20:READG$(H),B$(H,1):NEXT
2020 FORH=1TO27:READN$(H),N$(H):NEXT
2500 RETURN
2600 PLOT1,10,3:PLOT8,10,"W E L L   D O N E "
2601 SOUND1,0,0:SOUND1,0,0
2610 PLOT1,15,2:PLOT2,15,"You have solved this adventure"
2620 FORX=1TO10:FORY=255 TO 1 STEP-3:SOUND1,Y,10:SOUND1,255-Y,15:NEXTY,X
2630 SOUND1,0,0
3000 IFP%=29THENPRINTX$(7):P%=30:GOTO100
3001 IFP%=100THENPRINTX$(7):P%=101:GOTO100
3002 IFP%=13THENPRINT"I try to climb out, but slip back"
3003 IFP%=13THENPRINT"in the mud. I could be here forever!":GOTO100
3004 IFP%=60THENPRINTX$(7):P%=61:GOTO100
3005 IFP%=41THENPRINTX$(7):P%=40:GOTO100
3006 PRINT"I can't do that here!":GOTO100
3100 IFP%=30THENPRINTX$(8):P%=29:GOTO100
3110 IFP%=40THENPRINTX$(8):P%=41:GOTO100
3120 IFP%=101THENPRINTX$(8):P%=100:GOTO100
3130 IFP%=61THENPRINTX$(8):P%=60:GOTO100
3140 GOTO3006
3300 IFP%=74THENPRINTX$(27):P%=73:GOTO100
3310 IFP%=28THENP%=27:PRINTX$(27):GOTO100
3320 PRINT"I am not able to do that here idiot!":GOTO100
3360 IFP%=25THENPRINT"safe landing ". I landed in some soft sand":P%=35:GOTO100
3365 IFP%=63THENPRINT"S P L A S H":EXPLODE:P%=59:GOTO100
3400 IFP%=27THENP%=28:PRINTX$(27):GOTO100
3410 IFP%=73THENP%=74:PRINTX$(27):GOTO100
3420 PRINT"I can't do that here !":GOTO100
3500 IFP%=26THENPRINT"There isn't a bus in sight !":GOTO100
3510 IFP%=38THENPRINTX$(27):PRINT"I get off at the next stop":P%=26:GOTO100
3520 PRINT"DON'T BE STUPID !!!":GOTO100
3550 IFP%=90THENPRINTX$(27):P%=89:GOTO100
3555 IFP%=89THENP%=90:PRINT"I just made it":GOTO100
3560 IFP%=63THENPRINT"S P L A S H":EXPLODE:P%=15:GOTO100
3570 IFP%=59THENPRINT"SPLASH":EXPLODE:P%=15:GOTO100
3580 PRINT"I can't do that here!":GOTO100
3737 IFSE=1THENPRINTX$(25):GOTO100
4000 GOSUB5000
4010 IFL%=1THEN4030
4020 GOTO90
4030 EX=0
4040 FORH=1TO20:IFB$(H,1)=P%ANDB$(N$(R),1)=P%THENEX=1
4050 NEXT
4060 IFEX=0THEN90
4070 IFP%=103AND(R=1ORR=2) THENPRINTX$(19):GOTO100
4080 IFR=8THENSA=1
4090 IFR=18THENSB=1
4100 IFR=26THENSC=1
4103 IFR=25THENSB=1
4110 IFP%=64ANDR=7ANDSE<>1THENPRINTX$(3):GOTO100
4120 IFP%=64ANDR=23ANDSE<>1THENPRINT"I can't see it here!":GOTO100
4130 IFP%=36ANDR=27THENPRINT"I'm not that sort of person!":GOTO100
4140 IFP%=98ANDR=24THENPRINT"It's stuck":GOTO100
4150 IFR=23THENSS=1
4160 IFR=15ANDP%=23THENPRINTX$(11):SHOOT:GOTO100
4500 EX=0:FORD=1TO3:IFV$(D)=""THENV$(D)=G$(N$(R)):EX=1:D=5
4510 NEXT
4520 IFEX=0THENPRINT"I can't carry any more":GOTO90
4530 B$(N$(R),1)=0:GOTO90
5000 L$="":FORH=1TOLEN(Z$)
5010 IFMID$(Z$,H,1)= " "THENL$=RIGHT$(Z$, (LEN(Z$)-H)):H=H+40
5020 NEXT
5030 R=0:LX=0:IFLEN(L$)<2THENRETURN
5040 FORH=1TO27:IFLEFT$(N$(H),LEN(L$))=L$ THENLX=1:R=H
5050 NEXT
5060 RETURN
6000 GOSUB5000
6010 IFL%=1THEN6030
6020 PRINT"I can't see a ":L$:GOTO100
6030 EX=0
6040 FORD=1TO3:IFV$(D)=G$(N$(R)) THENV$(D)="" :EX=1
6050 NEXT
6055 IFEX=1THEN6070

```

```

6060 PRINT"I haven't got it":GOTO100
6070 B$(N$(R),1)=P%
6080 CLS
6090 IFR=26THENSC=0ELSEIFR=25THENSB=0ELSEIFR=8THENSA=0ELSEIFR=7THENSD=0
6100 IFR=22THENSS=0
6500 GOTO100
6600 PRINTCHR$(129):"I have :-"
6610 F%=0
6620 FORH=1TO3:IFV$(H)<>""THENPRINTV$(H):F%=1
6630 NEXT
6640 IFF%=0THENPRINT"not a sausage"
6660 GOTO100
6700 IFP%=64ORP%=35THEN6720
6710 PRINT"Don't be silly":GOTO100
6720 IFP%=64ANDSA<>1THENPRINTX$(15):GOTO100
6730 IFP%=35THENPRINT"The padlock is too rusty to unlock":GOTO100
6735 IFSE=0 THEN6740 ELSEPRINTX$(25):GOTO100
6740 SE=1:PRINTX$(29):PRINTX$(30):G$(16)="The £DIAMOND of Poglovia "1:GOTO100
6800 PRINTX$(27):PRINT"She smiles for a second":GOTO100
6900 IFP%=53ORP%=52THEN6910ELSEPRINT"I can't do that here":GOTO100
6910 IFP%=52THENPRINTX$(27):PRINT"I sail the boat":P%=53:B$(6,1)=53:GOTO100
6920 PRINTX$(27):PRINT"I paddle the boat":P%=52:B$(6,1)=52:GOTO100
7000 IFP%<>103THENPRINT"I can't give the diamond yet":GOTO100
7010 IFSS<>1THENX$(1)="I didn't have the diamond & He didn't like it hat!"
7020 IFSS<>1THEN60TO40000
7030 CLS:GOTO2600
7500 IFP%=23THENPRINT"Don't be so cruel!":GOTO100
7510 IFP%=36THENPRINT"I'm not a sadist !":GOTO100
7520 IFP%<>103THENPRINT"Not now !":GOTO100
7530 IFSB<>1THENPRINT"I haven't any weapon":GOTO100
7540 IFSC<>1THENPRINT"I PULL THE GUN !, but I have no bullet":WAIT100
7550 IFSC<>1THENX$(1)="I DON'T THINK HE LIKED THAT!":GOTO40000
7560 PRINT"I quickly load the gun and shoot":SHOOT
7570 WAIT200:CLS
7575 PLAY1,0,3,0:GOTO2600
10000 FORX=1TO2:PLOT1,X,14:PLOT2,X,X+2:PLOT8,X,"A QUESTION OF DEATH "1:NEXT
10005 PRINT"I am in a very difficult situation"
10010 PRINT"Whilst on holiday on the tiny island"
10020 PRINT"of Poglovia which is situated off the"
10030 PRINT"coast of mainland Zoblovia, I have"
10040 PRINT"experienced several disasters."
10050 PRINT"On the first day the heavy rains "
10060 PRINT"caused extensive flooding. Many areas"
10070 PRINT"of the island are still flooded."
10080 PRINT"I ran out of money two days ago and ,"
10090 PRINT"to cap it all, the mad EMPEROR OF"
10095 PRINT"ZOBLOVIA has invaded the island and"
10096 PRINT"Killed most of the inhabitants."
10100 PRINT"I have been told that he is hiding in"
10110 PRINT"a secret hideout somewhere on the "
10120 PRINT"the island and is planning to blow up"
10130 PRINT"the island if he does not get the "
10140 PRINT"the DIAMOND OF POGLOVIA ."
10150 PRINT"I must try to get the diamond to him"
10160 PRINT"or kill him before he blows the island!"
10170 PRINT"up. I have only 150 moves in which to do this."
10180 PRINT"GOOD LUCK"
10185 WAIT1000:REM CHANGE WAIT TIME TO GIVE MORE TIME TO READ INSTR UCTIONS
10190 RETURN
30000 DATAin Poglovia High Street,22,2,23,27,in the main shopping area,1,3,0,0
30010 DATAin the shopping precinct,2,4,8,7,outside the public library,3,5,0,6
30020 DATAin the civic centre,4,16,12,0,by the checkout desk of the library
30030 DATA0,0,4,86,in a small chemist's shop,0,0,3,0,on a main road ,0,0,9,3
30040 DATAon a main road,0,0,10,8,by some traffic lights,26,0,11,9
30050 DATAby a workman's hut. There is a hole in the ground SOUTH of here
30051 DATA0,0,0,10
30060 DATAat the entrance to some swimming baths,0,0,0,5
30070 DATAat the bottom of a large hole,0,0,0,0
30075 DATAat the shallow end of a swimming pool,12,0,0,0

```

```

30076 DATAswimming in the deep end,0,58
30080 DATA0,0,in the village square,5,0,0,17,outside the HOTEL POST
,0,21,16,18
30090 DATAon a narrow footpath leading towards the forest,0,0,17,1
9
30100 DATAin the forest. There is a tree I couldclimb here,0,0,18,0
30110 DATAat the top of a tree,0,0,0,0,in the hotel entrance,17,64,
0,0
30120 DATAin the high street,73,1,24,0,in an amusement arcade,24,0,
0,1
30130 DATAin a disused alleyway,0,23,25,22,on top of a low wall,0,0
,0,24
30140 DATAat a bus stop,0,10,0,0,outside Woolworth's,0,0,1,0
30150 DATAin the gardening department it is by the entrance
30155 DATA0,0,0,29,at the bottom of a staircase.
30160 DATA0,34,28,0,at the top of the stairs,0,0,31,0,in the electr
ical dept.
30170 DATA0,32,33,30,in the cafeteria,31,0,0,0,by a lighting displa
y,0,0,0,31
30180 DATAat the checkout,29,0,0,0
30190 DATAin a loading bay,0,0,36,0,in a narrow tunnel,0,37,0,35
30195 DATAat the end of a tunnel,36,0,38,0
30200 DATAat a bus terminus. There are plenty ofbuses here.,39,0,0,
37
30210 DATAat the edge of town,0,38,40,0
30220 DATAat some traffic lights. There is an open manhole here,4
2,0,0,39
30230 DATAat the bottom of a sewer. It is full of ffff",0,0,0,0
30240 DATAon a main road. The river nearby has burst its banks,45,
40,0,43
30250 DATApadding in about four feet of water.,44,0,42,0
30260 DATAoutside Barclays Bank. It is flooded,50,43,45,46
30270 DATAwhere the floods have swept the road North away.,0,42,0,
44
30280 DATAin the entrance to the bank. It is flooded,49,47,44,48
30282 DATAat the bank counter. There is no-one here !,46,0,0,0
30285 DATAin the main hall of the bank,0,0,46,0
30290 DATAin the managers office. It is flooded to a depth of 10 fe
et !!!
30300 DATA0,46,0,0,by a cinema. The flooding here is bad,52,44,51,0
30310 DATAinside the entrance of the cinema. It is shut!,0,0,0,50
30320 DATAin the road. The flooding to the Northprevents me from go
ing further
30330 DATA0,50,0,0,by the roof of a large flooded building,5
5,0,0,54
30340 DATAon a large plank of wood which is floating on the wat
er,0,0,53,0
30350 DATAon the roof of the building,0,53,56,0,by a chimney,0,57,0
,55
30360 DATAat the other side of the roof. The floods stretch for
miles.
30370 DATA56,0,0,0,on the steps,at the side of the pool,15,0,59,0
30380 DATAat the side of the pool,0,60,0,58,at the bottom of some s
teps
30390 DATA59,0,0,0,on a balcony,0,0,62,0,on a diving board,63,0,0,6
1
30400 DATAat the end of the board,0,63,0,0
30410 DATAin the reception,21,65,0,0,in the main hall,64,0,66,0
30420 DATAat the end of the hall. There are people dining here.
,0,67,0,65
30430 DATAat the end of a long bar,66,0,68,0,at the other end of th
e bar
30440 DATA0,69,0,67,in a lounge,68,71,70,72,in the toilets,0,0,0,69
30450 DATAat the end of the lounge,69,0,0,0,in an alcove,0,0,69,0
30460 DATAat the entrance to "PLOGVIA ICE CAVES",0,22,0,0
30470 DATAin a vast gloomy cavern. There is a guide here. He is D
EAD!
30471 DATA75,0,0,0
30480 DATAat the end of the gloomy cavern,85,74,76,80
30490 DATAin a wide passage,0,77,0,75,in a narrow passage,76,0,78,0
30500 DATAin the narrow passage. It twists roundhere,79,0,0,77
30510 DATAin the passage,88,78,0,0,by a subterranean waterfall,84,8
1,75,82
30520 DATAon a narrow ledge at the side of the waterfall,80,83,0,0
30530 DATApadding in the waterfall. I daren't go further.,0,0,80,
0
30540 DATAat a dead end,81,0,0,0,in a narrow passage.Ice blocks the
wayNorth
30550 DATA0,80,0,0,in a small cave,0,75,0,0,in the reference sectio
n,0,0,6,87
30560 DATAin a reading room,0,0,86,0,at the end of the gloomy passa
ge,89,79,0

```

```

30570 DATA0,in a small chamber. There is a narrow ravine to the N.,
0,88,0,0
30580 DATAon a narrow ledge. There is a passage South across the ra
vine.,91,0
30590 DATA0,0,in a passage deep inside the mountain,0,90,93,92
30600 DATAat a junction with passages in many directions,97,96,91
,0
30610 DATAin a tunnel of ice,94,95,0,91,in a vast underground chamb
er,0,93,0,0
30620 DATAin a chamber with a stalactite made ofice. It is enormous
,93,0,0,0
30630 DATAin a small office. It seems strange tofind an office here
!,92,0,0,0
30640 DATAin a sloping corridor. It's obviously man made,0,92,0,98
30650 DATAin a winding passage,0,99,97,0,in a corridor,98,100,0,0
30660 DATAby a sheer wall of ice. There is a rope hanging down h
ere.
30670 DATA99,0,0,0,at the top of the rope,0,0,102,0,in a corridor,0
,103,0,101
30680 DATAIN THE HALL OF THE MAD EMPEROR OF ZOBLOVIA. HE IS HER
E !!!!
30690 DATA102,0,0,0
34000 DATA"",it's far too dangerous,It's too heavy to lift
34010 DATAI'm sorry I don't speak the language,I'm sorry I can't un
derstand you
34020 DATAIt's stuck, I climb up,I climb down,I'm not hungry
34030 DATAthat word is not in my limited vocabulary !,AGGGHH
H it got me
34040 DATARubbing an electric lamp is stupid,I've heard that one be
fore
34050 DATAI'm nearly there
34055 DATAI don't have a key,I don't have a clue
34060 DATATime is running out,I'll try it if you like,I wouldn't re
commend it
34070 DATAWhat an ugly looking thing,not likely,HOW DARE YOU SPEAK
LIKE THAT
34080 DATAdon't be so rude,it said you are a fffff
34085 DATAit's already open dummy,it's locked!
34090 DATA0.K.,something happened,I can see something,It gleams bri
ghtly
34100 DATAwhat fun,It's too dangerous,It's too hard,I've nothing to
do it with
34110 DATAI've not got a shovel
35000 DATAThe EMPEROR OF ZOBLOVIA,103,a pint of beer,68,a bar of so
ap,70
35020 DATAa large suitcase,64,a key,65,a sailing dinghy,53,a padloc
k,35
35030 DATASome driftwood,54,a small puppy,23,some ice,85,a rusty na
il,16
35040 DATAa pair of swimming trunks,58,a nest,20,a cup of tea,32,a
table,66
35050 DATA"",64,a lever,98,a gun,47,a bullet,2
35060 DATAa small girl who looks very lost,36
36000 DATAEMPEROR,1,ZOBLOVIA,1,PINT,2,BEER,2,BAR,3,SOAP,3,SUITCASE,
4
36010 DATAKEY,5,SAILING,6,DINGHY,6
36015 DATAPADLOCK,7,LOCK,7,DRIFTWOOD,8,WOOD,8,PUPPY,9,ICE,10,NAIL,1
1
36017 DATATRUNKS,12,NEST,13
36020 DATACUP,14,TEA,14,TABLE,15,DIAMOND,16,LEVER,17,GUN,18,BULLET,
19
36030 DATAGIRL,20
40000 CLS:EXPLODE:FORX=1TO2
40004 PRINT:PRINT:PRINT:PRINT:PRINT
40010 PLOT1,X,14:PLOT2,X,X+1:PLOT3,X,"T H A T ' S   D O N E   I T "
40014 NEXT
40020 PRINT:PRINT:PRINTX$(1):PRINT:PRINT:PRINT
40030 PRINT"He pressed the button":EXPLODE
40040 PRINT:PRINT:PRINT:PRINTCHR$(134):"Do you want another game":
EXPLODE
40050 REPEAT:GET A$:UNTIL A$="Y"OR A$="N"
40060 IF A$="Y"THENRUN
40070 END

```

A Question of Death Listing

MCLOAD

By T. C. Brown

This is a routine for loading M/C without a lot of waste data statements. It works by requesting the max. number of bytes required then the start location.

The data is entered in a string (in hex code) where it is transferred to an array (MCE). There are various checks for errors built into the routine which (I hope) make it "MUG PROOF".

```
10 REM: Quick m/c load routine
20 REM: By T.C.Brown:Oct1983
30 CLS:PRINT:PRINT:PRINT
40 INPUT"MAX BYTES REQUIRED";MX
50 PRINT:INPUT"START ADDRESS";ST
60 DIMMC$(MX):CT=0' INSTRUCTION COUNTER
70 PRINTCHR$(17):CLS'DELETE CURSER
75 PRINT:PRINT
80 PRINTSPC(7)CHR$(148)CHR$(131)"Do not
use the '#key";CHR$(148)CH
R$(151)
90 PRINTSPC(7)CHR$(148)CHR$(131)"or spac
e symbol";CHR$(148)SPC(6)CH
R$(151)
100 PRINT:PRINTSPC(7)CHR$(148)"Hex code
only please";CHR$(151):PRIN
T:PRINT
110 PRINT:INPUTHX$
115 REM: load into array
116 PRINT"INSTRUCTION","ADDRESS","HEX CO
DE"
120 : FORC=1TOLEN(HX$)STEP2
130 : MC$(CTR)="#" + MID$(HX$,C,2)
140 : PRINT,CT,,HEX$(ST+CT),,,MC$(CT
)
150 : CT=CT+1:NEXT
200 REM: DATA CHECK
210 PRINTCHR$(144)CHR$(135)"Any errors";
CHR$(151)
215 REM: MUG TRAP
220 INPUT"(Y/N)";E$:IFLEN(E$)>1THEN220
230 IFE$="Y"THEN300
240 IFE$="N"THEN250 ELSE220
250 PRINT:PRINTSPC(7)CHR$(145)"Any more
data (Y/N)";CHR$(151)
260 INPUTD$:IF D$="Y"THEN100
```

```
270 IFLEN(D$)>1THENPING:GOTO260
280 IFD$="N"THENSHOOT:INPUT"Are you sure
";CH$
290 IFCH$="Y"THEN400ELSE250
300 REM: ERRORS
310 INPUT"Instruction number please";NUM
320 INPUT"New data please";DD$
330 MC$(NUM)="#" + DD$
340 GOTO210'CHECK FOR OTHER ERRORS
400 REM: DATA ENTRY FINISHED
410 PRINT"O.K"
500 REM: POKE INTO ADDRESSES
510 : FORP=0TOCT
520 : U=VAL(MC$(P))
530 : POKEST+P,U
540 : NEXT
600 PRINT"Any further instructions ?"
610 Q$=KEY$:IF Q$=""THEN610
620 IF Q$="N"THEN END
630 PRINT"Please enter new lines from 70
0"
```

.COMMENTS

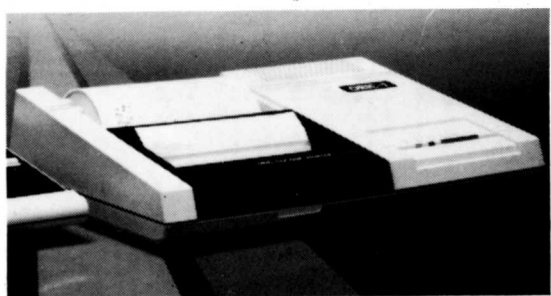
Lines 30 to 100 set up the screen
Lines 110 to 150 enter data
Lines 200 to 400 are data (& mug) checks
Lines 500 to 540 poke into the correct l
ocations
Lines 600 to 630 allow for further comma
nds:(E.G.'Doke')
Data entry is by hex code only in a stri
ng of upto 80 characters

COMPETITION!

Oric SuperCrossword

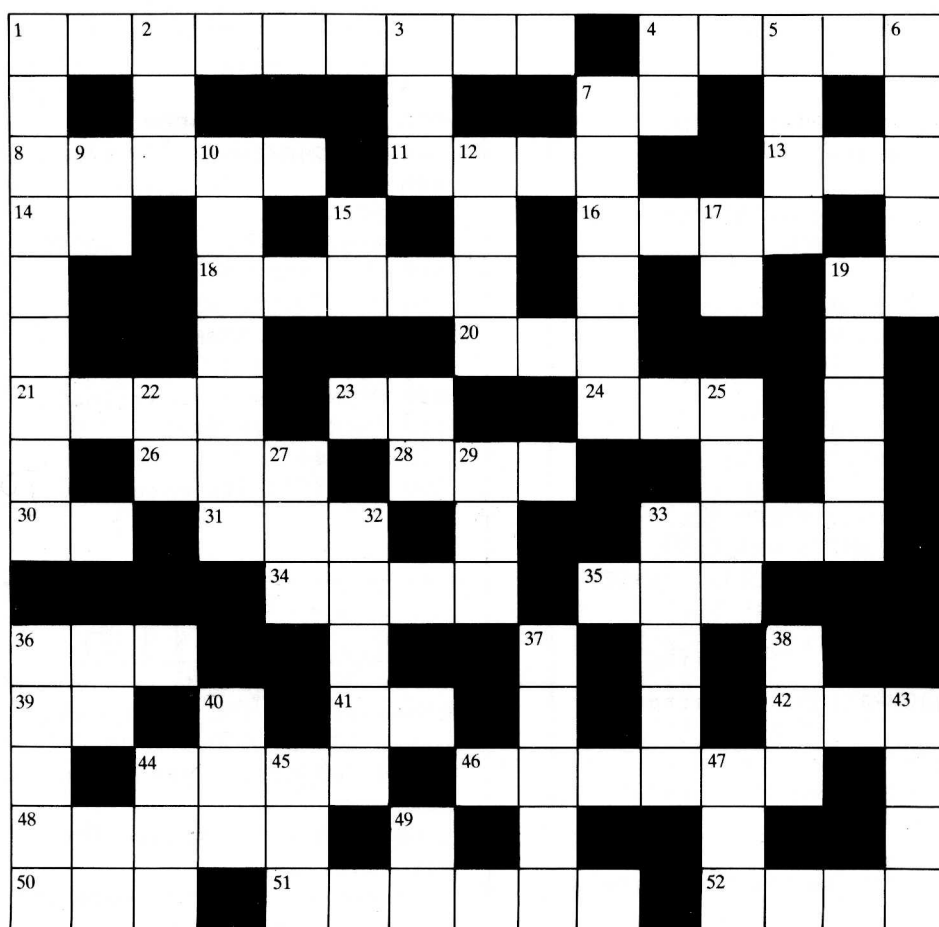
By Geoff Phillips

**1st PRIZE — AN MCP40 Oric
COLOUR PRINTER!**



**For 10 lucky runners-up
we have Defence Force
& Ultima Zone!**

cut along dotted line



HOW TO ENTER

Just complete the crossword and return it to: Unit 1, Techno Park, Newmarket Rd., Cambridge together with this completed caption:

I like Oric Owner because

.....

.....

(Not more than 20 words)

Closing date: 30 March 1984

Most answers have something to do with the Oric.

Clues: Across

1. You are what you read (4 5).
4. The late Arthur is a character set (5).
7. The location of character 64 (2).
8. The one byte number that represents a BASIC keyword (5).
11. Get the computer to twiddle its thumbs (4).
13. Getting a weasel out from a subroutine (3).
14. Conditional BASIC word? No – the mirror image! (2)
16. How many complements for the negative number? (4)
18. A question mark against this command (5).
19. Same as 7A (2).
20. The mythical creature steals an eye from the computer (3).
21. As well (4).
23. This number sounds edible! (2).
24. One of those ergonomic things (3).
26. A strangely uneven number (3).
28. The amount of memory sung by Roger Daltry (3)
30. Because (2).
31. This instruction does a lot of donkey work (3).
33. A priest might do this when a program goes wrong! (4).
34. Better not use FILL in this mode (4).
35. This ill-bred new Tansoft program gets things in the right perspective! (3).
36. A fruity version of Forth (3).
39. Same as 14A (2).
41. Out of this world! (2).
42. EE!! a powerful instruction (3).
44. A difficult feature to implement on the computer version of pinball! (4).
46. Start a loop after eating radishes (6).
48. Debug the outline of a program (5).
50. A curse on your number base (3).
51. Where they make garbage cans – two at a time! (6).
52. Two peeks are better than one! (4).

Down

1. Too much reading can cause an error (3 2 4).
2. Foreground (3).
3. A modern way of wiping out your program? (3).
4. Same as 7A (2).
5. What you wear for upper-case (4).
6. Get some data from your user (5).
7. Fight the rising edge! (6).
9. See 14A.
10. Noisy sound effect (7).
12. You don't need RUN with this transportable keyword (4).
15. See 23A.
17. Used in conditional rowing? (2).
19. Hip Hip! for all that data! (5).
22. Half of not bad (2).
25. 3 Feet behind the house (4).
27. Dorothy the Pixel. (3).
29. There's a lot of these to be splatted in a new Tansoft game. (3).
32. Actually the button connects to the NMI (5).
33. Get the computer to analyse a sentence (5).
36. This language did not come in the top three (5).
37. You can see right through these unwanted variables! (5).
38. You are allowed to omit or buy this keyword (3).
40. Hughey and Ralph, without 1024 – the answer is deliberate. (3).
43. Take a quick look at a memory location (4).
44. Levy the use of a mnemonic (3).
45. The source of London's power saying Zeppelin with a cold? (3).
47. Ampersand (3).
49. See 14A.

ORIC SUPERCROSSWORD © 1983 GMPHILLIPS

"CODE MASTER DELUXE"

By S. Mellin

'CODE MASTER DELUXE', is a game I have written for the 48K ORIC 1. As you will see it is based on the board game 'Super Mastermind'.

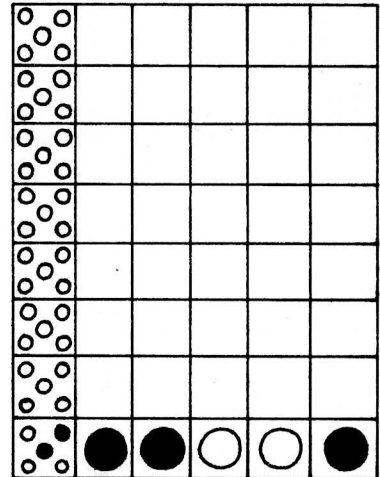
You will notice that the player does not pit his/her wits against 'ORIC' but against 'STEVE'. I decided to do this both because I was fed up with seeing 'Oric Wins' etc. etc. in programs and because I thought the game would be more enjoyable if it had a personal touch.

```

Y   G   B   BLU  R   O
1   2   3   4   5   6

GUESS NUMBER 1:  44615

GUESS NUMBER 2:
  
```



```

5 REM <<<< Code Master Deluxe >>>>
10 REM <<<< By Stephen J. Mellin >>>>
20 SP=0:SP$=STR$(SP)
25 SS=0:SS$=STR$(SS)
30 GOSUB 8000 'DEFINE CHARACTERS
35 GOSUB 5000 'TITLE PAGE
40 GOSUB 6000 'INSTRUCTIONS
45 PRINTCHR$(17);
50 GOSUB 7000 'DRAW BOARD
55 POKE616,5:PRINT:PRINTCHR$(17);
60 REM COMPUTE HIDDEN CODE
70 ZZ=Z+1
80 AF=INT(RND(1)*ZZ):A=AF
90 BF=INT(RND(1)*ZZ):B=BF
100 CF=INT(RND(1)*ZZ):C=CF
110 DF=INT(RND(1)*ZZ):D=DF
120 EF=INT(RND(1)*ZZ):E=EF
130 REM PLAYER'S INPUT
140 T=1:PRINT
150 IFPEEK(616)>25 THEN POKE616,6:PRINT:GOSUB1000
170 A=AF:B=BF:C=CF:D=DF:E=EF
180 PRINT"Attempt no. ";T
190 IFPEEK(616)>25 THEN POKE616,6:PRINT:Y9=1:GOSUB1000
200 INPUT TS
210 IFPEEK(616)>25 THEN POKE616,6:PRINT:GOSUB1000
220 LETS="RESIGN" THEN 3000 'END OF GAME
230 IFLEN(TS)>5 THEN PRINT"> Too long <";GOTO150
240 IFLEN(TS)<5 THEN PRINT"> Too short <";GOTO150
250 A1$=MID$(TS,1,1):A1=VAL(A1$):IF A1>Z THEN PRINT"> 1 is too big <";
GOTO150
255 IFASC(A1$)>57 OR ASC(A1$)<48 THEN PRINT"> Numbers only <";GOTO150
260 B1$=MID$(TS,2,1):B1=VAL(B1$):IF B1>Z THEN PRINT"> 2 is too big <";
GOTO150
265 IFASC(B1$)>57 OR ASC(B1$)<48 THEN PRINT"> Numbers only <";GOTO150
270 C1$=MID$(TS,3,1):C1=VAL(C1$):IF C1>Z THEN PRINT"> 3 is too big <";
GOTO150
275 IFASC(C1$)>57 OR ASC(C1$)<48 THEN PRINT"> Numbers only <";GOTO150
280 D1$=MID$(TS,4,1):D1=VAL(D1$):IF D1>Z THEN PRINT"> 4 is too big <";
GOTO150
285 IFASC(D1$)>57 OR ASC(D1$)<48 THEN PRINT"> Numbers only <";GOTO150
290 E1$=MID$(TS,5,1):E1=VAL(E1$):IF E1>Z THEN PRINT"> 5 is too big <";
GOTO150
295 IFASC(E1$)>57 OR ASC(E1$)<48 THEN PRINT"> Numbers only <";GOTO150
300 REM PLOT ATTEMPT ON BOARD
310 IF A1=7 THEN PLOT25,25-2*T,1:PLOT26,25-2*T,"#":GOTO330
320 PLOT25,25-2*T,A1:PLOT26,25-2*T,"#":GOTO330
330 IF B1=7 THEN PLOT27,25-2*T,1:PLOT28,25-2*T,"#":GOTO350
340 PLOT27,25-2*T,B1:PLOT28,25-2*T,"#":GOTO350
350 IF C1=7 THEN PLOT29,25-2*T,1:PLOT30,25-2*T,"#":GOTO370
360 PLOT29,25-2*T,C1:PLOT30,25-2*T,"#":GOTO370
370 IF D1=7 THEN PLOT31,25-2*T,1:PLOT32,25-2*T,"#":GOTO390
380 PLOT31,25-2*T,D1:PLOT32,25-2*T,"#":GOTO390
390 IF E1=7 THEN PLOT33,25-2*T,1:PLOT34,25-2*T,"#":GOTO410
400 PLOT33,25-2*T,E1:PLOT34,25-2*T,"#":GOTO410
410 PRINT"> SHALL I MARK? <";
420 GETZBS
430 IFZBS="Y" THEN 510
440 IFZBS="N" THEN 150
450 GOTO420
500 REM CHECK ATTEMPT
510 SB=0:SW=0
520 IF A1=ATHENSB-SB+1:A=-1:A1=-2
530 IF B1=ETHENSB-SB+1:B=-1:B1=-2
540 IF C1=CTHENSB-SB+1:C=-1:C1=-2
550 IF D1=DTHENSB-SB+1:D=-1:D1=-2
560 IF E1=ETHENSB-SB+1:E=-1:E1=-2
570 IF A1=ATHENSW-SW+1:A=-1:A1=-2
580 IF B1=ETHENSW-SW+1:B=-1:B1=-2
590 IF C1=CTHENSW-SW+1:C=-1:C1=-2
600 IF D1=DTHENSW-SW+1:D=-1:D1=-2
610 IF E1=ETHENSW-SW+1:E=-1:E1=-2
620 IF B1=CTHENSW-SW+1:C=-1:C1=-2
630 IF B1=DTHENSW-SW+1:D=-1:D1=-2
640 IF B1=ETHENSW-SW+1:E=-1:E1=-2
650 IF C1=ATHENSW-SW+1:A=-1:C1=-2
  
```

"Code Master Deluxe" Listing

```

660 IF C1=BTHENSW=SW+1:B=-1:C1=-2
670 IF C1=DTHENSW=SW+1:D=-1:C1=-2
680 IF C1=ETHENSW=SW+1:E=-1:C1=-2
690 IF D1=ATHENSW=SW+1:A=-1:D1=-2
700 IF D1=BTHENSW=SW+1:B=-1:D1=-2
710 IF D1=CTHENSW=SW+1:C=-1:D1=-2
720 IF D1=ETHENSW=SW+1:E=-1:D1=-2
730 IF E1=ATHENSW=SW+1:A=-1:E1=-2
740 IF E1=BTHENSW=SW+1:B=-1:E1=-2
750 IF E1=CTHENSW=SW+1:C=-1:E1=-2
760 IF E1=DTHENSW=SW+1:D=-1:E1=-2
800 REM PLOT MARKS
810 X=19
820 IFSB=0 THEN 870
830 FORN=1 TO SB
840 PLOTX,25-2*T,"^"
850 X=X+1
860 NEXTN
870 IFSW=0 THEN 920
880 FORN=1 TO SW
890 PLOTX,25-2*T,"%"
900 X=X+1
910 NEXTN
920 IFSB=5 THEN 3000 'END OF GAME
930 T=T+1
940 IFT>10 THEN 3000 'END OF GAME
950 GOTO 150
1000 REM CLEAR TEXT
1010 PRINT CHR$(17);
1020 FORM=1 TO 20
1030 PRINT " "
1040 NEXTM
1050 POKE 616,6:PRINT
1060 PRINT CHR$(17);
1070 IF Y9=1 THEN PRINT "Attempt no.";T
1080 Y9=0
1090 RETURN
2000 REM INCREMENT SCORE
2010 POKE 616,6:PRINT
2020 PRINT CHR$(17);
2030 GOSUB 1000
2040 PRINT CHR$(17);
2100 IFT$="RESIGN" THEN S1=0:S2=15:BS=0:V=0:GOTO 2300
2110 IFT>10 THEN S1=0:S2=12:BS=0:V=0
2120 IFT=10 THEN S1=1:S2=10:BS=0:V=0
2130 IFT=9 THEN S1=2:S2=9:BS=0:V=0
2140 IFT=8 THEN S1=3:S2=8:BS=0:V=0
2150 IFT=7 THEN S1=4:S2=7:BS=0:V=0
2160 IFT=6 THEN S1=5:S2=6:BS=1:V=1
2170 IFT=5 THEN S1=6:S2=5:BS=2:V=1
2180 IFT=4 THEN S1=7:S2=4:BS=3:V=1
2190 IFT=3 THEN S1=8:S2=3:BS=5:V=1
2200 IFT=2 THEN S1=9:S2=2:BS=8:V=1
2210 IFT=1 THEN S1=10:S2=0:BS=15:V=1
2300 PRINT "    SCORE"
2310 PRINT "    $$$$":PRINT
2320 PRINT "The score for":PRINT "that game was:"
2330 PLOT 1,12,"....."
2340 PLOT 1,12,NAS:PLOT 1,14+V,"STEVE....."
2350 S1$=STR$(S1):S2$=STR$(S2):BS$=STR$(BS)
2360 PLOT 17-LEN(S1$),12,S1$
2370 PLOT 17-LEN(S1$),12,1
2380 PLOT 17-LEN(S2$),14+V,S2$
2390 PLOT 17-LEN(S2$),14+V,1
2400 IFT$="RESIGN" THEN 2500
2410 IFT<7 THEN PLOT 1,13,"plus bonus of ..":PLOT 17-LEN(BS$),13,BS$
2420 IFT<7 THEN PLOT 17-LEN(BS$),13,1
2500 SP=SP+S1+BS:SS=SS+S2
2510 SP$=STR$(SP):SS$=STR$(SS)
2520 WAIT 100
2530 PLOT 15-LEN(SP$),2,SP$
2540 PLOT 15-LEN(SP$),2,1
2550 PLOT 15-LEN(SS$),3,SS$
2560 PLOT 15-LEN(SS$),3,1
2570 WAIT 500

```

```

2990 RETURN
3000 REM END OF GAME
3010 IFSB=5 THEN GOSUB 3100
3020 POKE 616,6:PRINT
3030 GOSUB 1000
3040 PRINT CHR$(17);
3050 IFSB<>5 THEN SOUND 1,1500,0:PLAY 1,0,1,2000
3060 PRINT ">> GAME OVER <<":PRINT:PRINT
3070 WAIT 300:IFSB=5 THEN 3200
3080 IFT$="RESIGN" THEN 3600
3090 IFT>10 THEN 3400
3100 PLAY 7,0,1,1
3110 FORN=1 TO 5
3120 MUSIC 1,4,1,9:MUSIC 2,3,5,9:MUSIC 3,3,8,9
3130 WAIT 6
3140 MUSIC 1,1,1,0:MUSIC 2,1,1,0:MUSIC 3,1,1,0
3150 WAIT 3
3160 NEXTN
3170 RETURN
3200 REM PLAYER CRACKED CODE
3210 IFT>4 THEN PRINT "Well done"
3220 IFT=4 THEN PRINT "EXCELLENT!"
3230 IFT=3 THEN PRINT "TREMENDOUS!!"
3240 IFT=2 THEN PRINT "INCREDIBLE!!"
3250 IFT=1 THEN PRINT "IT'S IMPOSSIBLE!!"
3260 WAIT 200:PRINT
3270 IFT=10 THEN PRINT "You just made it!"
3280 IFT<10 AND T>6 THEN PRINT "You did it in ";T
3290 IFT<7 AND T>2 THEN PRINT "You did it ":PRINT "in only ";T
3300 IFT=2 THEN PRINT "YOU DID IT IN TWO"
3310 IFT=1 THEN PRINT "YOU DID IT IN ONE"
3320 WAIT 200:PRINT
3330 IFT=2 THEN PRINT "YOU MUST BE A":PRINT "GENIUS,";NAS
3340 IFT=1 THEN PRINT "YOU MUST BE":PRINT "PSYCHIC,";NAS
3350 WAIT 200:PRINT:PRINT "THE HIDDEN CODE":PRINT "IS,OF COURSE....."
3360 GOSUB 3800
3365 PRINT
3370 IF Z<7 AND T<6 THEN PRINT "why not try":PRINT "skill level ";Z+1,";":PRINTNAS
3375 IF Z<7 AND T<6 THEN WAIT 200
3380 GOSUB 2000
3390 GOTO 4000
3400 REM PLAYER DID NOT CRACK CODE
3410 PRINT "Bad luck":PRINT
3420 PRINT "THE HIDDEN"
3430 PRINT "CODE IS....."
3440 GOSUB 3800
3450 IF Z=1 THEN 3490
3460 PRINT:PRINT "why not try":PRINT "skill level ";Z-1,";":PRINTNAS
3465 WAIT 200
3527 GOSUB 2000
3480 GOTO 4000
3490 PRINT:PRINT "why not try":PRINT "again,";NAS
3495 WAIT 200
3500 GOSUB 2000
3510 GOTO 4000
3600 REM PLAYER RESIGNED
3610 PRINT "THE HIDDEN"
3620 PRINT "CODE IS....."
3640 GOSUB 3800
3650 IF Z=1 THEN 3700
3660 PRINT:PRINT "why not try":PRINT "skill level ";Z-1,";":PRINTNAS
3670 WAIT 200
3680 GOSUB 2000
3690 GOTO 4000
3700 PRINT:PRINT "why not try":PRINT "again,";NAS
3710 WAIT 200
3720 GOSUB 2000
3730 GOTO 4000
3800 REM DISPLAY HIDDEN CODE
3810 A=AF:B=BF:C=CF:D=DF:E=EF
3815 WAIT 200
3820 PLOT 25,2,23:PLOT 25,3," "
3830 IFA=7 THEN PLOT 25,2,1:PLOT 26,2,"#":GOTO 3850
3840 PLOT 25,2,A:PLOT 26,2,"@"

```

[illegible]

```

5200 NEXTN
5210 WAIT20
5230 PLOT10,6," "
5240 PLOT10,7," "
5250 PLOT9,8,7
5260 PLOT10,8,"$$$$$$$$$$$$$$$$$$$"
5270 PLOT9,7,7
5280 PLOT10,7,CHR$(96)
5333 PLOT12,7,"Stephen J Mellin"
5300 WAIT200
5310 PRINTCHR$(17);
5990 RETURN
6000 REM INSTRUCTIONS
6010 CLS:INK7:PAPER4:PRINT
6020 PRINTCHR$(4);CHR$(27)"J CODE MASTER DELUXE"
6030 PRINT:PRINTCHR$(4)
6040 PRINT"DO YOU WANT INSTRUCTIONS?"
6050 PRINT:PRINT"IF YOU DO PRESS 'Y', IF NOT PRESS 'N'"
6060 GETX8$:IFX8$="Y"THENPRINTCHR$(17);:GOTO6090
6070 IFX8$="N"THENPRINTCHR$(17);:GOTO6620
6080 GOTO6060
6090 CLS:PRINT
6100 PRINTCHR$(4);CHR$(27)"J CODE MASTER DELUXE"
6105 PRINT:PRINTCHR$(4)
6110 PRINT"Hello,this is Steve,your opponent."
6115 PRINT:PRINT"The idea of this game is to break"
6120 PRINT"the hidden code which I will set"
6125 PRINT"at the beginning of each game."
6130 PRINT:PRINT"The code itself is a row of coloured"
6135 PRINT"markers hidden behind a shield at the"
6140 PRINT"top of the board."
6145 PRINT"Your task is to try and duplicate the"
6150 PRINT"exact colours and positions of the"
6155 PRINT"hidden code."
6160 PRINT"to do this place a row of coloured"
6165 PRINT"markers on the board.I will then"
6170 PRINT"indicate the correctness of your"
6175 PRINT"attempt.Use this information when"
6180 PRINT"considering your next attempt."
6185 PRINT:PRINT"PRESS SPACE BAR FOR MORE INSTRUCTIONS"
6190 GETX7$:IFX7$<>" "THEN6190
6195 CLS:PRINT
6200 PRINTCHR$(4);CHR$(27)"J CODE MASTER DELUXE"
6205 PRINT:PRINTCHR$(4)
6210 PRINT"you should enter your attempt as a"
6215 PRINT"row of five numbers e.g.01536."
6220 PRINT"Each number corresponds to a colour."
6225 PRINT"The colours in use will be displayed"
6230 PRINT"at the top of the screen with their"
6235 PRINT"corresponding numbers.The number of"
6240 PRINT"colours in use depends on which skill"
6245 PRINT"level (1-7) you select;the higher the"
6250 PRINT"skill level,the greater the number of"
6255 PRINT"colours in use."
6260 PRINT"Don't forget to press 'RETURN' to"
6265 PRINT"register your attempt."
6270 PRINT"When you have entered your attempt"
6275 PRINT"I will ask you whether you want it to"
6280 PRINT"be marked.If you do,press 'Y'.If you"
6285 PRINT"wish to change your attempt,simply"
6290 PRINT"press 'N' and enter your preferred"
6295 PRINT"combination for that attempt."
6300 PRINT"you have a maximum of ten attempts to"
6305 PRINT"break the hidden code.If at any time"
6310 PRINT"you wish to resign,enter 'RESIGN'"
6315 PRINT"as your next attempt."
6320 PRINT"PRESS SPACE BAR FOR MORE INSTRUCTIONS";
6325 GETX7$:IFX7$<>" "THEN6325
6330 CLS:PRINT
6335 PRINTCHR$(4);CHR$(27)"J CODE MASTER DELUXE"
6340 PRINT:PRINTCHR$(4)
6345 PRINT"The colours and corresponding numbers"
6350 PRINT"are as follows:"
6355 PRINT:PRINT" = Black (Bk)"
6360 PRINT"1 = Red (Rd)"

```



```

7030 REPEAT
7040 PLOT36,M,16
7050 PLOT18,M,23
7060 M=M+1
7070 UNTILM=26
7080 PLOT25,2,16:PLOT35,2,23
7090 PLOT25,3,16:PLOT35,3,23
7100 PLOT27,2,"HIDDEN"
7110 PLOT28,3,"CODE"
7120 FORM=2TO25
7130 PLOT24,M,"|"
7140 NEXTM
7150 PLOT25,1,"$$$$$$$$$$$":M=4
7160 REPEAT
7170 FORN=19TO34
7180 PLOTN,M,"$"
7190 NEXTN
7200 M=M+2
7210 UNTILM>25
7220 M=5
7230 REPEAT
7240 FORN=26TO34STEP2
7250 PLOTN,M,"#"
7260 NEXTN
7270 M=M+2
7280 UNTILM>23
7290 PLOT26,25,"1 2 3 4 5"
7300 PLOT19,25,"MARKS"
7310 CNS="#0=Bk 1=Rd 2=Gn 3=Ye 4=Bl 5=Ma 6=Cn 7=Em"
7320 FORP=#BB7FTO#BB7F+40
7330 POKEP,32
7340 NEXTP
7350 FORP=1TO4+5*Z
7360 POKE#BB80+P,ASC(MID$(CNS,P,1))
7370 NEXTP
7380 FORN=1TO4
7390 PLOT16,N,16:PLOT1,N,19
7400 NEXTN
7410 PLOT2,1,"$$$$$$$$$$$$$$$"
7420 PLOT2,4,"$$$$$$$$$$$$$$$"
7430 PLOT2,2,"....."
7440 PLOT2,2,NAS
7450 PLOT2,3,"STEVE....."
7460 PLOT15-LEN(SP$),2,SP$
7470 PLOT15-LEN(SP$),2,1
7480 PLOT15-LEN(SS$),2,SS$
7490 PLOT15-LEN(SS$),3,1
7990 RETURN
8000 REM DEFINE CHARACTERS
8010 FORP=0TO7
8020 READP1
8030 POKE46592+P,P1
8040 NEXTP
8050 DATA12,30,63,63,63,63,30,12
8060 FORP=0TO7
8070 READP2
8080 POKE46360+P,P2
8090 NEXTP
8100 DATA12,18,33,33,33,33,18,12
8110 FORP=0TO7
8120 READP3
8130 POKE46368+P,P3
8140 NEXTP
8150 DATA0,0,0,63,0,0,0,0
8160 FORP=0TO7
8170 READP4
8180 POKE46376+P,P4
8190 NEXTP
8200 DATA0,0,12,18,18,12,0,0
8210 FORP=0TO7
8220 READP5
8230 POKE46832+P,P5
8240 NEXTP
8250 DATA0,0,12,30,30,12,0,0
8990 RETURN

```

Features

Oric Copy

The enclosed program is a simple text screen dump for the 48K Oric using the “!” key facility for extension commands to provide for a copy command. The program was written using a Tandy CGP-115 printer/plotter, but as this is basically the Sharp 4-colour printer/plotter in disguise, there is no reason why it should not work for the other badge engineered printers like Oric’s own MCP 40. Owners of other printers should however, check and change if necessary, locations Hx044A and Hx044F to include control codes which govern Carriage Return (CHR\$ 13) and Line Feed (CHR\$ 10) respectively for their own printers.

The way the program works is simplicity itself. When “!” is entered, the first text screen address is loaded into locations Hx0408 and Hx0409. This forms the base address, and its contents are loaded into the A register. This is then printed by making a Call to HxF57B which is the Print-the-character-in-A routine in ROM. 16K owners should therefore change location Hx0466 to Hx3B which is the MSB of the first address of the 16K text screen. The LPRINT routine in ROM is the same for both models. The Index register Y is incremented, and the offset added to the base address and the print process repeated. When 40 characters

have been printed (i.e. 1 line), a Carriage Return and Line Feed are issued to the printer and the X register decremented. This is initialised at 27, which is the number of lines on the screen. The base address is then incremented by 40; (start of the next line), and the whole process repeated until the X register is decremented to zero. The locations storing values for the X and Y registers are initialised and a Return made to BASIC.

Control codes and attributes affecting the screen pose no problem as they do not affect the printer. The ones that do are primarily cursor controls and as the Keyboard scan is turned off anyway to prevent the interrupts causing random printing of CHR\$ 126 and CHR\$ 127, the issue does not even arise. The exception is CTRL-D which when used with the ESC character gives double-height printing. As the double-height characters are the same characters stacked on top of each other, the printer will merely print them twice.

The program is stored in locations Hx0400 to Hx046A; and despite what the Oric manual says, I have found that Page 4 is in fact quite safe under any condition even after NEW and Reset. It is initially stored in BASIC in DATA statements as shown in the listing, and is POKEd into place by the

Loader. The DATA statements should be entered exactly as shown, but for clarity, I have included an assembler listing with line/line commentary. RUN the BASIC program when it has been entered, and the “!” can then be used to copy the screen. It will only fail if the text screen is moved, for example when RELEASE is used or in the HIRES mode.

```
1 REM Extension command for
  COPY using “!”. D. Yan 1983
10 DATA001BEAEAEAA000
  B9000008C0004207BF5
  AC0004C8C028D0
  EF204904 AE0104
  CA8E0104 F015
20 DATA18AD080469288D
  080490D6AD09041869018D
  09044C0504204904A9008D
  00044A91B8D0104
30 DATA60A90D207BF5A90
  A207BF56020
  CAE62005042004
  E860EAEA
40 DATAA9A88D0804A9BB8
  D09044C5404,Z
99 REM BASIC Loader
100 :A=#400:READD$
110 :REPEAT:FORX=
  1TOLen(D$)STEP2
120 :D=VAL(“#”+
  MID$(D$,X,2))
130 :POKEA,D:A=A+1:
  NEXT
140 :READD$:UNTILD$=
  “Z”
149 REM Enable “!”
150 :DOKE#2F5,#460
```

Features

Oric Copy

Assembler listing. ORIC COPY

	<i>HX</i>	<i>LOC</i>	<i>CODE</i>	<i>MNEMONIC</i>	
YSTORE	0400	00			Character count store.
XSTORE	0401	1B			Line count store.
	0402	EA		NOP	
	0403	EA		NOP	
	0404	EA		NOP	
CCOUNT	0405	A0	00	LDY Hx00	Initialise character count.
FBYTE	0407	B9		LDA SOR,Y	Character at base address SOR offset
SOR(LSB)	0408	00			Y into A register.
SOR(MSB)	0409	00			
	040A	8C	00 04	STY YSTORE	Put Y register in YSTORE.
	040D	10	7B F5	JSR LPRINT	LPRINT character in A.
	0410	AC	00 04	LDY YSTORE	Retrieve character count.
	0413	C8		INY	Next character.
	0414	C0	28	CPY Hx28	40 characters? (1 line).
	0416	D0	EF	BNE FBYTE	Not end of line, get next byte.
	0418	20	49 04	JSR EOL	End of line.
	041B	AE	01 04	LDX XSTORE	Get line count.
	041E	CA		DEX	1 line finished.
	041F	8E	01 04	STX XSTORE	Put back updated value.
	0422	F0	15	BEQ END	27 lines done?
	0424	18		CLC	No, prepare for next line.
	0425	AD	08 04	LDA SOR(LSB)	Get LSB of base address.
	0428	69	28	ADC Hx28	Update base address to next line.
	042A	8D	08 04	STA SOR(LSB)	Put back updated value.
	042D	90	D6	BCC CCOUNT	No carry, so back to CCOUNT.
	042F	AD	09 04	LDA SOR(MSB)	Carry set, so get MSB.
	0432	18		CLC	Prepare for update.
	0433	69	01	ADC Hx01	Add carry bit to MSB.
	0435	8D	09 04	STA SOR(MSB)	Put back updated value.
	0438	4C	05 04	JMP CCOUNT	All values updated, back to CCOUNT
END	043B	20	49 04	JSR EOL	Reset print head to margin.
	043E	A9	00	LDA Hx00	Reinitialise YSTORE.
	0440	8D	00 04	STA YSTORE	
	0443	A9	1B	LDA Hx1B	Reinitialise XSTORE.
	0445	8D	01 04	STA XSTORE	
	0448	60		RST	End of END routine, go to
					STARTEND.
EOL	0449	A9	0D	LDA Hx0D	Prepare to send Carriage Return.
	044B	20	7B F5	JSR LPRINT	Send C/R to printer.
	044E	A9	0A	LDA Hx0A	Prepare to send Line Feed.
	0450	20	7B F5	JSR LPRINT	Send L/F to printer.
	0453	60		RST	End of EOL routine.
STARTB	0454	20	CA E6	JSR HxE6CA	Turn off keyboard scan.
	0457	20	05 04	JSR CCOUNT	Perform program.
STARTEND	045A	20	04 E8	JSR HxE804	Turn on keyboard scan.
	045D	60		RST	Return to BASIC.
	045E	EA		NOP	
	045F	EA		NOP	
STARTA	0460	A9	A8	LDA HxA8	LSB first byte of address of Text
					screen.
	0462	8D	08 04	STA SOR(LSB)	Value into SOR.
	0465	A9	BB	LDA HxBB	MSB first byte of address of Text
					screen.
	0467	8D	09 04	STA SOR(MSB)	Value into SOR.
	046A	4C	54 04	JMP STARTB	Go to STARTB.

Simple Simon

By S. W. Lucas

Rules

The game is a test of memory in which the computer displays a random sequence of coloured blocks in the four corners of the screen (accompanied by an appropriate sound).

Your task is to reproduce the sequence using the keys:-

R Y

C B

There is also a time limit placed on your response and this may be adjusted at the start of the game.

Explanation of the Program

line 1:- selects text mode and releases the hires memory – so that it will work in 16K

line 4:- sets the score (SC) to zero

line 50:- selects subroutine for the titles

lines 55–56:- move the cursor to correct position

lines 57–58 select colours used

lines 70–76 waits for you to enter 1 or 2 in response to the difficulty and then choose X%

to alter the difficulty adjust line

76 – the smaller the value of X% the harder the game will be!
lines 80–86 set the messages
lines 90–1000 contain the main control of the program:- this consists of a REPEAT UNTIL loop lines 110 and 340

the condition causing it to jump out of the loop is when $AC \neq 0$ and this is set to 1 when an incorrect response or a slow response occurs

lines 5000–5999:- subroutine to select a random letter which is added to the end of B\$– the sequence

lines 6000–6600:- subroutine to display the random sequence

lines 7000–7200:- subroutine to display your attempt

lines 10000–10100:- subroutine

for the red box

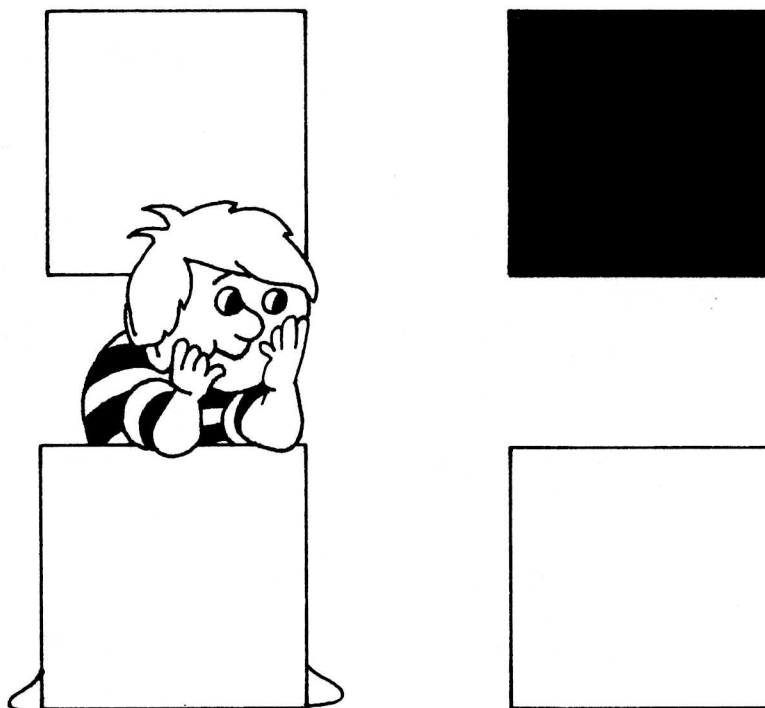
lines 20000–20100:- subroutine for green box

lines 30000–30100:- subroutine for yellow box

lines 40000–402100:- subroutine for blue box

The above 4 subroutines create the boxes in TEXT made by setting the attributes on each line to the appropriate colour and then later in the line setting them back to white:- the background colour chosen by PAPER in line 57

lines 50000–50340 contain the subroutine for the titles and lines 50100–50110 select whether the instructions are displayed (lines 50120 onwards).



Simple Simon Listing

```

1 TEXT:GRAB
4 SC=0
5 REM SC=SCORE
10 REM THIS IS A VERSION OF THE FAMILIAR GAME OF "SIMON"
15 DIM M$(6):REM MESSAGES
20 REM < C > 1983 S.W. LUCAS
30 REM FOR THE ORIC 1 48K :REMOVE REM STATEMENTS FOR THE 16K VERSIO
N
40 TEXT
50 GOSUB50000:REM SUBROUTINE FOR TITLES AND INSTRUCTIONS
55 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
56 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
57 PAPER 7:REM SET BACKGROUND TO WHITE
58 INK 1:REM SET FOREGROUND TO RED
60 PLOT1,10,18:PLOT2,10,"ENTER THE DIFFICULTY LEVEL 1 OR 2"
69 PRINTSPC(10);
70 REPEAT:REM ROUTINE TO CHECK FOR THE PRESSING OF 1 OR 2
72 GETZ$
74 UNTIL (Z$="1" OR Z$="2")
75 LET Z=VAL(Z$)
76 IFZ=1 THEN XZ=40 ELSE XZ=20
80 REM MESSAGES
81 M$(1)="WATCH"
82 M$(2)="NOW REPEAT"
83 M$(3)="CORRECT"
84 M$(4)="WRONG !"
85 M$(5)="THE CORRECT SEQUENCE WAS:--"
86 M$(6)="TOO SLOW !"
90 REM THE FOLLOWING SECTION IS THE MAIN CONTROL SECTION OF THE GAM
E
95 SC=0
96 AC=0
100 B$="":REM B$ HOLDS THE RANDOM CHOICE OF LETTERS TO CREATE THE S
EQUENCE
110 REPEAT
200 GOSUB5000:REM ADD A LETTER TO THE SEQUENCE
205 REM NOW PRINT WHETHER RIGHT, WRONG OR RAN OUT OF TIME
250 GOSUB6000:REM ROUTINE TO DISPLAY THE SEQUENCE
300 GOSUB7000:REM SUBROUTINE FOR YOUR TRY
306 IF TI > XZ*4 THENCLS:PLOT15,13,M$(6):AC=1:GOTO450
307 REM TOO SLOW
310 IFAAS<>MID$(B$,ZX,1)THENCLS:PLOT15,13,M$(4):AC=1:GOTO400
311 REM GOTO IT WRONG
320 IFZX=LEN(B$)THENCLS:PLOT15,13,M$(3)ELSECLS:PLOT15,13,"PROGRAM H
AS CRASHED!"
325 REM GOT IT RIGHT NOW INCREMENT THE SCORE
330 SC=SC+1
340 UNTILAC<>0
399 END
400 WAIT50:PLOT10,13,M$(5)
410 WAIT50
415 GOSUB6000
420 WAIT50:CLS:PLOT15,10,"YOU SCORED "
421 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRI
NT:PRINT

```

```

422 PRINT:PRINT:PRINTSPC(18);SC
425 PRINT:PRINT:PRINT"PRESS <SPACE> FOR ANOTHER GAME"
426 REPEAT :GETA$:UNTILA$=" "
430 RUN
450 UNTIL TI > XZ*4
451 WAIT50
452 CLS
455 WAIT50:PLOT15,10,"YOU SCORED "
456 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRI
NT:PRINT
457 PRINT:PRINT:PRINTSPC(18);SC
458 PRINT:PRINT:PRINT"PRESS <SPACE> FOR ANOTHER GAME"
459 REPEAT
460 GETA$
470 UNTILA$=" "
480 RUN
1000 END
4999 REM SUBROUTINE TO SELECT A RANDOM LETTER AND ADD IT TO B$
5000 LET A=INT(RND(1)*4)+1
5005 A$=""
5010 IFA=1 THENA$="R"
5020 IFA=2 THENA$="Y"
5030 IFA=3 THEN A$="C"
5040 IF A=4 THEN A$="B"
5050 IFA$=""THENPRINT"PROGRAM HAS CRASHED":REM ERROR CHECK
5060 LET B$=B$+A$:REM ADD THE RANDOM LETTER TO B$
5999 RETURN
6000 CLS:REM SUBROUTINE TO DISPLAY THE RANDOM SEQUENCE
6010 PLOT15,13,M$(1)
6020 WAIT50:REM SELECT VALUE TO SUIT YOU
6025 LETXY=LEN(B$):REM CHECKS LENGTH OF RANDOM SEQUENCE
6030 XX=1
6035 REPEAT
6040 C$=MID$(B$,XX,1)
6050 IFCS="R"THENS=1ELSEIFCS="Y"THENS=2ELSEIFCS="C"THENS=3ELSEIFCS=
"B"THENS=4
6060 LET S=S*10000:REM S=SUBROUTINE FOR COLOUR
6070 GOSUB S
6080 XX=XX+1
6500 UNTIL(XX>XY)
6600 RETURN
7000 CLS:REM SUBROUTINE FOR YOUR TRY
7010 PLOT15,13,M$(2):WAIT50:REM SELECT WAIT VALUE TO SUIT YOU
7015 LET TI=0
7020 CLS
7035 ZX=0
7040 REPEAT
7045 LET TI=TI+1:AA$=KEYS:REM TI IS A MEASURE OF THE TIME RESPONSE
7050 IF (AA$="R"ORAA$="Y"ORAA$="C"ORAA$="B") THEN7060
7051 IFTI>XZ*4 THENGOTO7060:REM CHECK FOR TOO SLOW RESPONSE TIME
7055 GOTO7045
7060 ZX=ZX+1
7061 IFAAS="R"THEN GOSUB10000

```


Simple Simon Listing

```

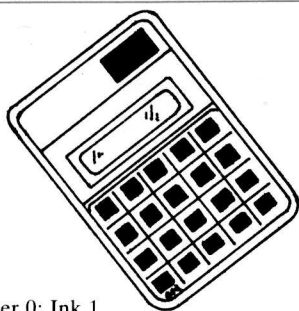
7062 IFAAS="Y" THEN GOSUB 20000
7063 IFAAS="C" THEN GOSUB 30000
7064 IFAAS="B" THEN GOSUB 40000
7070 UNTIL (ZX=LEN(B$) OR AAS<>MID$(B$,ZX,1) OR TI>XZ*4):REM ADJUST
    TO SUIT
7200 RETURN
9998 REM SUBROUTINE TO DRAW THE RED BOX
9999 REM AND MAKE THE APPROPRIATE SOUND
10000 CLS
10005 FORY=2 TO 12
10010 PLOT5,Y,17:REM SET THE ATTRIBUTE TO RED
10020 PLOT16,Y,23:REM AND THEN SET IT BACK TO WHITE
10030 NEXT Y
10040 REM NOW MAKE THE NOTE
10050 MUSIC1,2,1,15
10060 REM XZ= TIME DELAY ADJUSTED WHEN YOU CHOOSE THE SPEED
10070 WAIT XZ
10080 SOUND1,0,0:REM THEN TURN SOUND OFF
10100 RETURN
9998 REM SUBROUTINE FOR DRAWING THE GREEN BOX
9999 REM AND MAKING THE APPROPRIATE SOUND
20000 CLS
20005 FOR Y= 2 TO 12
20010 PLOT23,Y,18:REM SET ATTRIBUTES TO GREEN
20020 PLOT34,Y,23:REM AND BACK TO WHITE
20030 NEXT Y
20040 REM NOW FOR THE SOUND
20050 MUSIC1,3,1,15
20060 REM XZ= TIME DELAT WHICH IS ADJUSTED AT THE START
20070 WAIT XZ
20080 SOUND1,0,0:REM TURN OFF SOUND CHANEL
20100 RETURN
29998 REM SUBROUTINE FOR DRAWING THE YELLOW BOX
29999 REM AND MAKING APPROPRIATE NOTE
30000 CLS
30005 FORY=14 TO 24
30010 PLOT5,Y,19:REM SET ATTRIBUTES TO YELLOW
30020 PLOT16,Y,23:REM AND BACK TO WHITE
30030 NEXT Y
30040 REM NOW FOR NOTE
30050 MUSIC1,4,1,15
30060 REM XZ= DELAY
30070 WAIT XZ
30080 SOUND1,0,0:REM TURN SOUND OFF AGAIN
30090 RETURN
39998 REM BLUE BOX SUBROUTINE
39999 REM INCLUDING SOUND
40000 CLS
40005 FOR Y=14 TO 24
40010 PLOT23,Y,20:REM SET ATTRIBUTES= BLUE
40020 PLOT34,Y,23:REM AND BACK TO WHITE
40030 NEXT Y
40040 REM AND THEN THE SOUND
40050 MUSIC1,5,1,15
40060 REM XZ= TIME DELAY ADJUSTED AT THE START

```

```

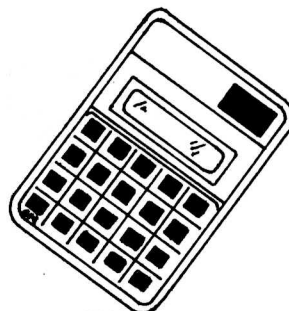
40070 WAIT XZ
40080 SOUND 1,0,0:REM TURN OFF THE SOUND AGAIN
40100 RETURN
49999 REM SUBROUTINE FOR THE TITLES
50000 PAPER0:INK6
50005 CLS
50010 PLOT9,2,14:REM SET ATTRIBUTES FOR DOUBLE HEIGHT FLASHING
50020 PLOT10,2,3:REM SET ATTRIBUTES=YELLOW
50030 PLOT11,2,"SIMPLE SIMON"
50040 PLOT9,1,14:REM SET TO DOUBLE HEIGHT FLASHING
50050 PLOT10,1,1:REM SET TO RED
50060 PLOT11,1,"SIMPLE SIMON"
50070 PLOT10,10,"<C> 1983 by S.W. Lucas"
50080 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:P
RINT.
50085 PRINT:PRINT:PRINT:PRINT:PRINT
50090 PRINT:PRINT:PRINT"Do you want instructions <Y>es <N>o ?";
50095 REM NOW CHECK FOR PRESSING Y OR N KEYS
50100 REPEAT
50105 GET AS
50107 UNTIL (AS="Y" OR AS="N")
50110 IF AS="N" THEN CLS:RETURN:REM DON'T PRINT INSTRUCTIONS
50120 PLOT1,5,3:REM SET TO YELLOW
50130 PLOT2,5,"This is a game of memory. Watch the"
50140 PLOT1,6,3:REM YELLOW
50150 PLOT2,6,"screen while coloured blocks flash in"
50160 PLOT1,7,3:REM YELLOW
50170 PLOT2,7,"a random sequence in the 4 corners of"
50180 PLOT1,8,3:REM YELLOW
50190 PLOT2,8,"the screen."
50200 PLOT1,15,5:REM MAGENTA
50210 PLOT2,15,"You have to reproduce the sequence"
50220 PLOT1,16,5:REM MAGENTA
50230 PLOT2,16,"USING THE KEYS:"
50240 PLOT1,18,7:REM WHITE
50250 PLOT2,18," R Y
50260 PLOT1,20,7:REM WHITE
50270 PLOT2,20," C B
50280 PRINT:PRINT:PRINT:PRINT:PRINT"Don't take too long or you will
lose !"
50290 PRINT:PRINT"Press any key to continue"
50300 REM WAIT FOR ANY KEY TO BE PRESSED
50310 GETAS
50320 CLS
50340 RETURN

```



Maths

By M. Shaer



```

1. Paper 0: Ink 1
2. Rem *****
3. Rem *By M. D. Shaer*
4. Rem *****
5. CLS
10 Print:Print:Print
20 Print " MATHS! ! "
25 Print " 666 "
30 Print:Print
35 Print "This program is made to help you with"
40 Print "your Maths. The computer will give you"
50 Print "10 questions on each of the following"
60 Print "subjects."
70 Print:Print
85 Prints "Addition"
90 Print
100 Print "Subtraction"
110 Print " and"
120 Print "Multiplication"
150 Print:Print
160 Print "Press any key to continue."
A$ = KEY$
200 A$ = Key: If A$ = " " Then 200
250 CLS
300 Print:Print:Print
350 Print "INSTRUCTIONS"
400 Print "-----"
500 Print:Print
510 Print "1. + = addition"
520 Print "2. - = subtraction"
530 Print "3. * = multiplication"
600 Print:Print:Print
650 Print "now then what questions do you want"
700 Print "to be given."
750 Print:Print
800 Print "1 = addition"
850 Print "2 = subtraction"
900 Print "3 = multiplication"
1000 Input A
1100 If A = 1 then 2000
1200 If A = 2 then 3000
1300 If A = 3 then 4000
1400 If A<>1 then 1000
1500 If A<>2 then 1000
1600 If A<>3 then 1000
2000 For N = 1 to 10
2050 CLS
2100 Print:Print:Print:Print:Print:Print
2150 A = int (Rnd (1) * 500) +1
2200 B = int (Rnd (1) * 300) +1
2250 Print A: Print "+": Print B
2300 Input Q
2350 If Q=A+B then 2500
2400 If Q<>A+B then 2700
2500 Print "CORRECT!!"
2525 M=M+1
2550 Ping
2575 Print "Press any key for the next sum": A$=KEY$
2600 A$=KEY$: If A$=" " then 2600
2625 Next N: Goto 250
2650 End
2700 Print "WRONG THE ANSWER WAS!!!"
2750 Print A+B
2800 ZAP
2850 Print "Press any key for the next sum": A$=KEY$

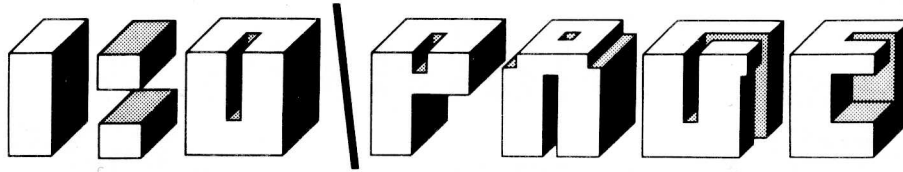
```

```

2900 A$=KEY$: if A$=" "Then 2900
2950 Next N: Goto 250
2975 END
3000 for W=1 TO 10
3050 CLS
3100 Print:Print:Print:Print:Print:Print
3150 c=Int(Rnd(1)*500)+1
3200 d=Int(Rnd(1)*300)+1
3250 Print c: Print "-": Print d
3300 Input L
3350 If L=c-d Then 3500
3400 If L<>c-d Then 3700
3500 Print "CORRECT!!!"
3550 M=M+1
3600 PING
3625 Print "Press any key for the next sum": A$=KEY$
3650 A$=KEY$: If A$=" "Then 3650
3675 Next W:Goto 250
3690 END
3700 Print "WRONG THE ANSWER WAS!!!"
3705 Print c-d
3750 ZAP
3775 Print "Press any key for the next sum": A$=KEY$
3800 A$=KEY$: If A$=" "Then 3800
3850 Next W: Goto 250
3900 END
4000 For S=1 TO 10
4050 CLS
4100 Print:Print:Print:Print:Print:Print
4150 E=Int(Rnd(1)*100)+1
4200 F=Int(Rnd(1)*50)+1
4250 Print E: Print "*": Print F
4300 Input Y
4325 If Y=E*F Then 4450
4350 If Y<>E*F Then 4650
4450 Print "CORRECT!!!"
4500 M=M+1
4525 PING
4550 Print "Press any key for the next sum": A$=KEY$
4575 A$=KEY$: If A$=" " Then 4575
4600 Next S: Goto 5000
4625 END
4650 Print "WRONG THE ANSWER WAS!!!"
4700 Print E*F
4725 ZAP
4750 Print "Press any key for the next sum": A$=KEY$
4800 A$=KEY$: If A$=" " Then 4800
4850 Next S: Goto 5000
5000 CLS:Print:Print:Print "Do you want to;"
5050 Print:Print
5100 Print "1=Have More Sums"
5150 Print
5200 Print "2=End The Game"
5250 Print
5300 Print "3=Know Your Score"
5400 input Z
5500 If Z=1 Then Goto 5
5600 If Z=2 Then 6000
5700 If Z=3 Then 6500
5750 If Z<>1 Then 5400
5800 If Z<>2 Then 5400
5850 If Z<>3 Then 5400
6000 CLS: END
6500 CLS:Print:Print:Print:Print "Your Total Was: "; M "Right"

```

Regulars



Dear Sirs,

The PONTOON program in Issue 2 is excellent, and has given us many hours of delight.

There is a problem, however. The computer allows a player to 'stick' on any number (although in normal play, we would not allow a player to 'stick' below 16).

To overcome this, I suggest an additional line:

```
1005 IF T < 16 AND TA < 16
THEN CLS: GOTO 1040
```

This will not give the option of another card if the total is below 16 but will go direct to the request for bet value or twist.

R. Sibthorpe

Editor: Anyone else got some improvements?

Dear Sir,

The attached example program from Oric Owner issue 4, was found to need the marked up modifications to obtain satisfactory operation. An 'RTS' instead of a 'BRK' instruction was more useful for experimentation with one off program runs (line 1F2E).

E. P. Hollister

Mistake, should not have been printed

Example

	1F00	A9 00	LDA	#\$0
	1F02	A0 9A	LDY	#\$9A
	1F04	20 ED D3	JSR	GIVAYF
	1F07	20 DD DE	JSR	MOVAF
	1F0A	A9 00	LDA	#\$0
	1F0C	A0 05	LDY	#\$5
	1F0E	20 ED D3	JSR	GIVAYF
	1F11	20 E3 DD	JSR	FDIVT
A0 —	1F14	A9 1F	LDY	#\$1F
	1F16	A2 F0	LDX	#\$F0
	1F18	20 A5 DE	JSR	MOVMF
	1F1B	20 D1 E0	JSR	FOUT
	1F1E	EA	NOP	
	1F1F	EA	NOP	
0 —	1F20	EA	NOP	
1F22 E8 INX —	1F21	A2 01	LDX	#\$01
5 —	1F23	BD 00 01	LDA	\$100,X
	1F26	F0 06	BEQ	\$1F2E
	1F28	9D 10 BE	STA	\$BE10,X 1
	1F2C	D0 F5	BNE	\$1F23
	1F2E	00	BRK	
		60	RTS	
?		BD 00 01		

Arcade sounds

By P. Kohlenbrander



This program enables you to compose your own Arcadetype tunes and jingles to use in your game programmes. If you want to use this little program in a games program then just delete the last three lines and line 30 and change line 20 to J\$= "number sequence you composed" and line 40 the L in the WAIT statement to the number you typed in when the computer asked for the tonelength when you were composing.

To make a special note longer just type the appropriate number in several times, e.g. (16445399999). For the tones use keys 1-9 and for tonelength use +/-2 for those very quick tunes and +/-15 for the jingle-type tunes.

```
5 CLS
10 PLAY 7,0,1,1000 (can be changed to your own wishes)
20 INPUT "Type in the tune by using keys 1-9
   (max. 80 char.)"; J$
30 INPUT "Type in tonelength (0>)"; L
40 FOR T=1 TO LEN(J$)
50 M=ASC(MID$(J$,T,1))-48
60 MUSIC 1,3,M,9
70 WAIT L
80 NEXT T
90 PLAY 0,0,0,0
100 PLOT3,5,"Play it again?(Y/N)"
110 GETA$:IF A$<>"Y" THEN 120 ELSE 40
120 END
```

Quickie: – ‘Calendar’

By Martyn J. Hall

This is a program which will print up each month of the year choosen, separately on the screen. Then it will give you the chance to continue or end the program. If you continue then it

will print up the next month and it will do this up to December when it will ask you if you want another go or not. On the other hand if you decide to end the

program part way through then it will go to the choice routine to see if you want another go or not. At this point it will either end completely or start again.

The Main Ideal of the program

The main idea is that the user can pick out any day in a certain month and year, and get a cal-

ender of that year or as much as the user wants i.e. starting from January. He/she can then do it

for as many times as he/she wants, from the year 0001 onwards.

Listing

```
1 GOSUB 10000
5 CLS
10 DIM D(12),W$(7),DPM(12),MO$(12)
20 GOSUB 1000
30 PRINT "Please enter the first two digits of the year":PRINT
40 PRINT "i.e.1900- - - - ->19(FIRST TWO DIGITS)";
49 C=0
50 REPEAT
55 C=C+1
60 GET A$
70 IF ASC(A$)<48 OR ASC(A$)>57 THEN 60
75 M$=M$+A$
80 PLOT 29+C,4,A$:PRINT CHR$(9)
90 UNTIL C=2
95 PRINT
100 REM XXXXXXXXXXXXXXXXXXXXXXXX
110 REM XX  DATE ROUTINE      XX
120 REM XXXXXXXXXXXXXXXXXXXXXXXX
130 REM
140 C=0
150 PRINT"Please enter the date in the following form :—"
155 PRINT:PRINT SPC(8),"DAY  MONTH  YEAR"
160 PRINT:PRINT SPC(9),"07   05   83"
165 J$(1)="":J$(2)="":J$(3)=" "
166 YE$="":A$=""
170 PRINT
190 REM
195 TT%=0
200 FOR F=1 TO 3
210 FOR G=1 TO 2
220 GET A$
230 IF ASC(A$)<48 OR ASC(A$)>57 THEN 220
240 J$(F)=J$(F)+A$
250 NEXTG
255 PLOT13+TT%,14,J$(F)
256 TT%=TT%+6
260 NEXTF
270 MO=VAL(J$(2)):DT=VAL(J$(1)):YE$=M$+J$(3):YE=VAL(YE$)
271 IF YE=0 THEN GOSUB 5000:GOTO 100
275 IF YE/4=INT(YE/4)THEN DPM(2)=29
280 IF MO<1 OR MO>12 OR MO<>INT(MO) THEN GOSUB 5000:GOTO 100
```


Prime Numbers By B. Woodland

Function

To determine all prime numbers up to value Z. In this example Z=200. Values X and Y may be altered for the most pleasing display with other values of Z.

Rationale

The array B(n) is seeded with 2, the first prime. Each value of A is divided by each prime, in ascending numerical order, then residing in B(n). If a clean division results, the number is not prime, and not saved in B(n). If $A < \#$ the square of the prime

deviser, then further division is redundant – that number is prime, and stored in the next space in B(n). Finally, all primes are displayed.

```
10 Z=200: DIMB(Z/2):
   B(1)=2
20 C=1: A=2: X=12: Y=24
30 CLS: PRINT
   SPC(9);CHR$(27)“L
   PRIME SEARCH”
40 FOR D = 1 TO C
50 IF A/B(D)=INT(A/B(D))
   THEN 90
60 F=B(D)*B(D): IF F>A
   THEN 80
70 NEXT
```

```
80 C=C+1: B(C)=A
90 A=A+1: IF A<=Z
   THEN 40
100 CLS: PAPER 0: PRINT
   CHR$(17)
110 PLOT 8,0,
   CHR$(7)+“PRIMES UP
   TO “+STR$(Z)
120 E=2: D=8: FOR A = 1
   TO C
130 PLOT D,E,STR$(B(A))
140 E=E+1: IF E>Y THEN
   D=D+X: E=2
150 NEXT
160 GET AK$: Paper 7:
   PRINT CHR$(17): END
```

‘Pattern’ By Martyn J. Hall

Program Description

This is a short program which after being initially started it will draw circles around an ellipse shape and then the background will be shaded in different colours. The product of all this being that you will have an ellipse made of circles so that there is a hole in the middle with a shaded background.

Program Listing

```
10 PAPER 1
20 CLS
30 PRINT “PLEASE WAIT
   FOR A WHILE”:
   WAIT400
40 HIRES
```

```
50 INKO
60 AS=“PATTERN: – by
   Martyn J. Hall”
70 CURSET,40,10,3
80 FOR N=1 TO LEN(AS)
90 CHAR ASC(MIDS(AS,
   N,1)),0,1
100 CURMOV6,0,3
110 NEXTN
120 A=70: B=50
130 FOR L=0 TO
   2*3.141592654 STEP .1
140 CURSET120+COS(L)*A,
   100+SIN(L)*B,1
150 CIRCLE 30,1
160 NEXTL
170 FOR P=41000 TO 48960
   STEP40
180 Y=INT(23*RND(1))+1:IF
```

```
Y 17 THEN 180
190 POKEP,Y
200 NEXTP
210 END
```

NB: By changing what A equals in line 120 the length of the ellipse can be changed, and by changing what B equals then that will alter the height of the ellipse. Also by changing the step in line 130 you can alter the density of the picture i.e. a very small number will produce a black ellipse with a hole in the centre whereas a fairly small whole number will produce large shaped holes due to the circles.

Oric Quickies

By D. R. Thwaites

LINES

Draws lines on the screen under control of the cursor keys. HIRES mode is used and each press of a cursor draws a line 5 pixels long. The current position is marked by a "+" sign. The "1" key exits the program and the "\ " key clears the screen for a new drawing.

```
5 REM Program 1 - Lines
100 HIRES
110 X=120 : Y=100 : CURSET X,Y,0
120 GOSUB 400
130 GET A$
140 IF A$="\ " THEN PING : GOTO 100
150 IF A$="1" THEN ZAP : TEXT : END
160 GOSUB 400
170 A=ASC(A$)-7
180 ON A GOSUB 200,200,300,300
190 GOTO 120
200 : B=10*(A-1.5)
210 : IF X+B<5 OR X+B>234 THEN 299
220 : DRAW B,0,1
230 : X=X+B
299 : RETURN
300 : B=10*(3.5-A)
310 : IF Y+B<5 OR Y+B>194 THEN 399
320 : DRAW 0,B,1
330 : Y=Y+B
399 : RETURN
400 : CURMOV -2,-3,3
410 : CHAR 43,0,2
420 : CURMOV 2,3,3
499 : RETURN
```

Is similar but draws character sized blocks on the screen instead of lines. It works in TEXT mode and the four cursor keys draw one block at a time in the appropriate direction. The current cursor position is marked by a dotted, rather than solid, block. The "1" and "\ " keys have the same functions as in program 1.

BLOCKS

It is straightforward to define further operations in the 'LINES' program to draw diagonals e.g. using the four keys just above the cursor keys.

Blocks gives two modes : DRAW (set at startup) which draws lines or blocks and RUBOUT which deletes lines or blocks. These are selected with the "D" and "R" keys.

```
5 REM Program 2 - Blocks
100 CLS
110 X=18 : Y=13 : PLOT X,Y,126
120 GET A$
130 IF A$="\ " THEN PING : GOTO 100
140 IF A$="1" THEN ZAP : CLS : END
150 A=ASC(A$)
160 IF A= 8 THEN B=-1 : C= 0 : GOTO 300
170 IF A= 9 THEN B= 1 : C= 0 : GOTO 300
180 IF A=10 THEN B= 0 : C= 1 : GOTO 300
190 IF A=11 THEN B= 0 : C=-1 : GOTO 300
200 GOTO 120
300 IF X+B<1 OR X+B>36 THEN 120
310 IF Y+C<1 OR Y+C>25 THEN 120
320 PLOT X,Y,126
330 X=X+B : Y=Y+C
340 PLOT X,Y,126
350 GOTO 120
```

These programs are in their simplest forms. Some improvements are obvious, e.g. PAPER and INK commands for coloured drawings. A further simple addition is to allow rubbing out of previously drawn lines. This could be accomplished by the following additions/changes.

LINES

```
105 P=1
152 IF A$="D" THEN P=1
154 IF A$="R" THEN P=0
220 : DRAW B,0,P
320 : DRAW 0,B,P
```

BLOCKS

```
105 P=128
152 IF A$="D" THEN P=128
154 IF A$="R" THEN P=32
320 PLOT X,Y,P
340 PLOT X,Y,126
```

Subscription

Why you should subscribe to ORIC OWNER

Your Oric 1 is one of the most powerful low-cost computers on the market today. Oric Owner is the only magazine totally dedicated to the Oric and its supporting hardware and software.

From issue to issue Oric Owner is able to bring you in-depth information on the Oric, advance news on the latest add-ons, superb programs to try and 'behind the scenes' interviews with the engineers who designed it.

If you wish to keep ahead of the latest developments and news on the Oric then Oric Owner is essential. At only £10 for a year's subscription of 6 issues (£15 for overseas readers) can you afford to be without Oric Owner?

SUBSCRIBE TODAY!

subscription form

Please send me the next 6 issues of
Starting from issue

ORIC OWNER

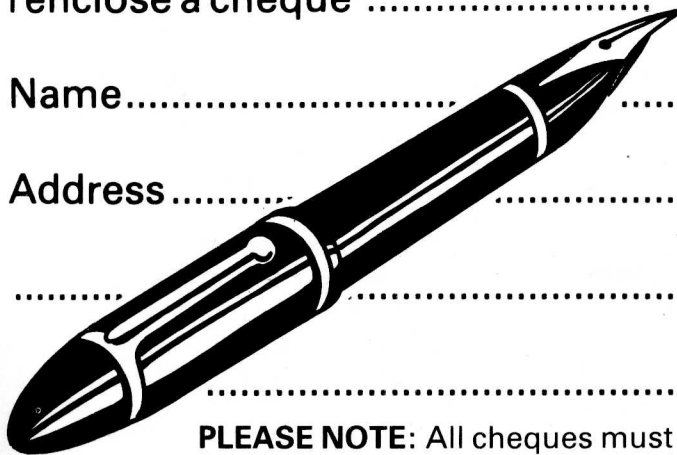
I enclose a cheque for £10

Name

Address

.....

.....



PLEASE NOTE: All cheques must be made payable to: Tansoft Ltd, Unit 1 & 2, Techno Park, Newmarket Road, Cambridge

Cassette File Manager

By R. L. Laygo

Here is a useful Utility Program for the serious Programmer who does not own a disk drive yet, but would like to be able to handle program files more efficiently using the ordinary cassette.

Ideally, a cassette drive with the auto Program search facility will help one skip through programs with more ease than having to verify by sound. Good cassette drives also allow for better resolution of data at the 2400 baud transfer rate (fast mode).

There is no more need to buy expensive 5-minute cassettes, as even a good quality C-60 will store numerous programs compactly. This MENU program is reserved or sectored at the start of every side with 000-035 count (fast mode) allowing sufficient space for updating.

You might also want to put files into separate volumes according to the following classification:

1. PROGRAM FILES

1.1 Self-authored Utilities, Games, etc.

1.2 Commercial Software backups

2; LITERAL ASCII FILES

2.1 Machine Code data

2.2 Screen data

2.3 Text data

3. Others

Delete lines 10-40 in actual use. Lines 100-999 heading the list file makes things easier at the time of updating. After creating or updating the MENU, always Autosave by typing: CSAVE "MENU—SIDE A (or B)", AUTO

```
10 REM: ***CASSETTE FILE
  MANAGER***
20 REM: BOYET LAYGO
  VER 2.0/2.0/10SEP83/
  MANAMA
30 REM
40 REM 98Ff
99 HIMEM #8FF-
100 FILE$="ORIC-1
  VOL. 1, UTILITIES"
110 FT$="CASSETTE FILE,
  Contents:"
120 DATA 000-035 "MENU—
  SIDE A"
130 DATA PROGRAM
  FILE. . .
999 DATA END OF FILE
1000 MC$="AD009F482033F4
  EEC19F68C90DD0
  F1A9008DC19F60":
  ADDVEC=#FC0:
  GOSUB2000 9Fc0
1010 CLS: PAPER0: INK1:
  RESTORE
1020 PRINT FILE$: PRINT:
  PRINT: PRINT FT$:
  PRINT
1030 :45 NOT PO THEN 1050
  :if PO=-1 THEN 1050
```

```
1040 :L$=FILE0+CHR$(10)
  +CHR0(13): GOSUB
  3000 $
1042 :L$=CHR$(15)+FT$+
  CHR$(14)+CHR$(10)+
  CHR$(13): GOSUB 3000
1050 REM: FILE CONTENTS
1060 PC=0 :REPEAT
1070 READ P$: PRINT P$
1080 :IF NOT PO THEN 1090
  ELSE L$=P$: GOSUB
  3000
1090 PC=PC+1 :UNTIL P$=
  "END OF FILE" OR
  PC=18
1100 PING: IF P$="END OF
  FILE" THEN 1130
1110 PRINT:PRINTCHR$(27);
  "Lpress any key to
  continue": GET A$
1120 CLS: GOTO1050
1130 PRINT: PRINT
  CHR$(27); "LPRINTOUT
  REQUIRED";: INPUT
  R$
1140 IF LEFT$(R$,1) <>"Y"
  THEN 1160
1150 PO=TRUE: GOTO 1010
```

```
1160 PAPER1: INK0: PING
1170 INPUT "REPEAT"; R$:
  IF LEFT$(R$,1) <>"Y"
  THEN 1190
1180 PO=FALSE: GOTO 1010
1190 CLS: PRINT "ENJOY
  YOUR
  PROGRAMMING!"
1999 END
2000 REM: LOADING
  M-CODE
2010 FOR I=1 TO LEN(MC$)
  STEP2
2020 :MC=VAL("#"+MID$
  (MC$,I,2))
2030 :POKE ADDVEC+
  INT(I/2+A),MC
2040 NEXT I
2050 RETURN
3000 REM: LPRINT
3010 FOR I=1 TO LEN(L$)
3020 :POKE #9F00+I-1,
  ASC(MID$(L$,I,1))
3030 NEXT I
3040 :POKE #9F00+I-1, 13
3050 CALL #9FC0: RETURN
```

Software

HERE IS AN EXAMPLE OF
A PRINTOUT WHICH
STORES NEATLY WITH
THE CASSETTE VOLUME

ORIC-1 VOL. 1, UTILITIES	365-390 "ROTO-ONE"	S	SIDE B
CASSETTE FILE,Contents:	AUTO		
000-035 "MENU—SIDE A"	393-398 "ROTO-ONE"	000-017 "LPRINT3"	A#9F00
037-087 FREE	AUTO	E#9FFF S	
089-143 "PERSPECTIVE 3"	400-406 "ROTO-ONE/1"	019-025 "LPRINT VER1.3"	
147-176 "PERSPECTRANS1"	AUTO	027-034 "LPRINT/BASIC"	
S AUTO	407-414 "ROTO-ONE/2"	036-045 "TEXTFORM"	
180-205 "PERSPECTRANS2"	AUTO	047-058 "TEXT"	
S AUTO	417-424 "ANIMATION1"	060-072 "TEXT3"	
208-220 "DRAWA" S AUTO	AUTO	074-085 "TEXT V1.3"	
222-237 Unclassified	428-434 "ANIMATION2"	087-098 "TEXT"	
241-259 "DRAW2"	AUTO	100-111 "TEXT 4"	
289-309 Unclassified	436-441 "BASICSUBR"	113-124 "TEXTV2"	
310-352 "TRANSTWO" S	443-455 "LPRINT VER1.0" S	126-138 "TEXT V2.1"	
AUTO	AUTO	140-152 "TEXT GOOD"	
354-360 "TRANSTWO"	457-469 "LPRINT VER1.1" S		
AUTO	AUTO		
	470-666 FREE		

For 48K £6.95 inc.

SUPER M/C ARCADE GAME FROM SELECTED BRANCHES

- Skill Levels, Full Colour, Platforms, Ladders, Fire-balls, Rolling Barrels, Umbrellas, Hearts, Hall of Fame, Sound Effects.

Your won't see a better game for the Oric.

*Author – Adrian Sheppard.

For 48K £6.95 inc.

REAL M/C ARCADE ACTION

- 4 Screens, Skill Levels, Full Colour, Road, Cars, Lorries, River, Crocs, Logs, Hall of Fame, Sound Effects.

Perspire your way up the screen.

*Author –
Adrian Sheppard.



SEVERN SOFTWARE

For 48K £8.50 inc.

- Assembler/Disassembler/Editor
- Handles full 6502 mnemonics. Features Save & Reload M/C Assemblies & Disassembles at any address.

You can't program seriously without Oricade.

*Author – Adrian Sheppard.

For 48K £6.95 inc.

Exciting Graphic Adventure

Where in the Castle Perilous is the Holy Grail? Gather armour and weapons to fight monsters. Sell treasure to a trader in exchange for strength potions and wound ointment. Where will the warp take you to? This is a test of skill, luck, logic & intelligence.

For 48K £6.95 inc.

A challenging adventure set in the mines of Moria.

- Can you survive encounters with the monsters of Middle-Earth? Will the wizard help you? Are you fated to die beside the sealed doors? Or have you the power to open them? Unless you find Durins Ring you will never leave the mines alive!

*Sold at branches of LASKYS and all good software shops – ask for SEVERN SOFTWARE.

Dealer/Distributor enquiries welcome – excellent trade terms.

Please send me your.....program.
For my Oric-1

I enclose cheque/P.O. for £.....

NAME.....

ADDRESS.....

..... Post Code.....

SEVERN SOFTWARE

OO 5, School Crescent, Lydney, Glos. GL15 5TA.

ORIC-1 SOFTWARE COMPARE THESE PRICES

XENON-1	IJK	£7.50
INVADERS	IJK	£6.75
REVERSE	IJK	£5.95
FANTASY QUEST	IJK	£5.95
3D MAZE & BREAKOUT	IJK	£6.75
CANDYFLOSS & HANGMAN	IJK	£6.75
ZODIAC	A&F	£6.20
DEATH SATELLITE	A&F	£6.20
THE ULTRA	PSS	£6.40
LIGHT CYCLE	PSS	£6.40
CENTPEDE	PSS	£6.40
THE GAUNTLET	PSS	£6.40
INVADERS	PSS	£6.40
HOPPER	PSS	£6.40
ORICMON	PSS	£7.95
FORTH	TANSOFT	£13.50
ORIC BASE	TANSOFT	£13.00
CHES	TANSOFT	£8.95
ORIC MUNCH	TANSOFT	£7.00
RAT SPLAT	TANSOFT	£7.00
DEFENCE FORCE	TANSOFT	£7.00
ORIC CALC	TANSOFT	£13.00
FLIGHT	TANSOFT	£4.00
THE HOBBIT	TANSOFT	£13.50
AUTHOR	TANSOFT	£13.00

ALL PRICES INCLUDE VAT & P&P - ALLOW 21 DAYS FOR DELIVERY
PLEASE SUPPLY

TOTAL ENCLOSED £

NAME

ADDRESS

SEND ORDERS TO:- POSTAL COMPUTER SERVICES,
FREEPOST P.O. BOX 2, MIDHURST, WEST SUSSEX, GU29 9BR.

ORIC GAME FREAKS

IF YOU WANT JOY... STICK TO US



SPECTRAVIDEO™

For further details
phone 01-203 6366/7
and ask for David Bishop.
Also available at the PCW Show

VULCAN
ELECTRONICS LTD

The UK's leading
Joystick Distributor

ORIC-1

FOUR QUALITY PROGRAMS FROM SUPERIOR SOFTWARE



LUNAR MISSION (48K) £7.95

"I'll have two hundred"

Jeff Brown, Centresoft, Tipton.



URUGUAY

CAPITAL?
?MONTEVIDEO
CORRECT
POPULATION?

WORLD GEOGRAPHY (48K) £7.95

This effective program builds a firm knowledge of country names, populations and capitals, through a series of graded question and answer sessions.

An attractive high resolution map dominates the display, with the session's text covering the bottom of the screen. A cursor associates location with country name.

Performance ratings help assess progress through each session, allowing the direction of further work.

ASSEMBLER (16K/48K) £9.95

A powerful tool for machine code programmers, with flexible editing facilities.

Features: Labels any length, simple arithmetic on operands, numbers may be specified in binary, octal, decimal or hex, source code may be saved to, and loaded from, tape. Available directives: OPT, ORG, EQU, RES, DEFB, DEFW, DEFS.

Supplied with clear and comprehensive documentation. PLEASE CIRCLE MEMORY SIZE WHEN ORDERING

TOOLKIT (16K/48K) £9.95

A high quality programmers' aid, for use on its own, or as a (co-resident) complement to the ASSEMBLER. The TOOLKIT includes a full disassembler, allowing output to memory and printer, a monitor, to step through machine code one instruction at a time, SEARCH, COPY and VERIFY. Disassembler output may be edited and reassembled (with ASSEMBLER), or stored on tape. Full documentation supplied. PLEASE CIRCLE MEMORY SIZE WHEN ORDERING

Clip & post to: Darkstar, FREEPOST, LEEDS, LS1 1YY.

Please send:

Lunar ☐ 48K Name
Mission ☐ 48K Address
World ☐ 48K
Geography ☐ 48K
Oric ☐ 16K/48K
Assembler ☐ 16K/48K
Oric ☐ 16K/48K
Toolkit ☐
Dealer ☐
Kit ☐

Cheque/P.O. enclosed £ payable to Darkstar. Overseas orders add 15%.

* Dealer kits available:
Phone 0532 452223 or tick box

* More quality programs required.

48 hour despatch by first class post

All prices include VAT and P. & P.

ORIC-1 USERS CLUB

Join Now and receive:

- A Free 'Oric Toolbox' Cassette
- Monthly Members' Newsletter
- Details of our 'Star Tape of the Month'
- Our Mail Order Catalogue with the widest Oric range at 10% discount.

ANNUAL MEMBERSHIP£5.00
6 MONTHS TRIAL£3.00
OVERSEAS (1 YEAR)£8.00

To join send fee, name, address and
Tel. No. or 50p for our Mail Order
Catalogue only

Members deduct 10% from any item
advertised and add p&p per item: 30p
Software; £8 Hardware. Overseas add
extra £1 per item.

LATEST ORIC RELEASES

Available NOW

Centipede	£6.95
Speech Synthesizer	£57
Xenon I	£8.95
Chess	£9.95
Oric Trek	£9.95
The Hobbit	£14.95
Database	£15.00
Hopper	£6.95
Oric-Munch	£7.95
Zodiac Adventure	£9.95
Games Comp. I	£7.95
Zorgon's Revenge	£8.95
Word Processor	£15.00
Oric Forth (+Man.)	£14.95
Dinky Kong	£6.95
Dig Dog	£6.90
Painter	£6.95
Harrier Attack	£7.95
Fantasy Quest	£6.50
Space Crystal	£6.95

ORIC 1 USERS CLUB

31 North Street, Wareham, Dorset BH20 1AD
Telephone: 09295 51383 Telex: 41112 OLYTRI G

with your Access, Barclaycard, American Express numbers for
immediate despatch (24hr. Service)

ORIC PERIPHERALS

NOW IN STOCK

MCP40 4 COLOUR PRINTER/PLOTTER
£149.95 incl. cable

SHINWA CP80 PRINTER, F and T
Feeds, True Descenders, Graphics,
Centronics - ideal for Oric, Dragon,
BBC, etc.£322.00

Both Printers also ideal for Dragon,
BBC, etc. P.A.S.E. JOYSTICK CONVER-
TER - Takes any digital Joysticks,
including free games tape ..£14.95

QUICK SHOT JOYSTICK£14.95

All Prices include VAT and
FREE DELIVERY



ORIC BASE

Ovacam have changed their
name to Oric Base and moved
from 70 Water Street,
Chesterton, Cambridge to
2 THE TECHNO PARK,
NEWMARKET ROAD,
CAMBRIDGE, CB5 8PB.
Their new Telephone No is:
Teversham (02205) 2352

Please send all repairs with
detailed letters explaining the
problems in full to the new
address above.

Calling all Advertisers!



The Oric Owner is the only magazine
devoted to the Oric-1 computer.

Virtually every person who buys an Oric
will get this magazine. If you are selling
Oric related products can you afford to
miss this chance to reach the market you
are aiming for?

For full details of our advertising rates
contact Carolyn on Teversham 2261 and see
if she can help you.

**NEW
RELEASES**

THE ULTRA

THE ULTRA

You have been challenged to a duel to the death in outer space by the ULTRA, an evil race of mutants from the darkest depths of the galaxy. We dare you to accept their challenge!

This is a PURE arcade game and a must for all you alien zappers! 16 different screens make things really tricky.

ORIC 1 16K 48K

LIGHT CYCLE

The MASTER CONTROL PROGRAM has ordered you to race your light cycle on the infamous grid. This is an incredibly fast, adrenalin pumping game that is very addictive. There is an option to race against either another player or the computer – but we warn you its very very good!

ORIC 1 16K 48K



**FOR FURTHER INFORMATION RING
COVENTRY (0203) 667556**

